International Journal of Design & Nature and Ecodynamics
Vol. 15, No. 1, February, 2020, pp. 13-23
Journal homepage: http://iieta.org/journals/ijdne

Dimension Reduction for Network Systems Using Structure Model Aggregation

Valeriy Gorbachov[1], Dmytro Sytnikov[1], Oleg Ryabov[2*], Abdulrahman Kotaeba Batiaa[1], Olha Ponomarenko[1]

[1] Kharkiv National University of Radio Electronics, Rybalko street, Kharkiv 61082, Ukraine
[2] National Institute of Advanced Industrial Science and Technology, Tokyo 135-0064, Japan

Corresponding Author Email: oleg.r@aist.go.jp

https://doi.org/10.18280/ijdne.150103

Received: 7 October 2019
Accepted: 12 February 2020

Keywords:
aggregation, element connection scheme, large scale system, structure model reduction, UML diagram

ABSTRACT

Analysis of real network systems with high accuracy leads to building complex high-dimension models. When the time for determining system, parameters plays an important role the system modeling, time becomes a critical parameter. So, one needs to find an approximate simpler representation of such network systems while preserving the characteristic properties of the higher dimension system. In this paper, an approach related to formal transformations of the system structure model using multilevel aggregation has been applied to obtain a reduction in computational complexity and faster modeling. Software implementation of the developed algorithm allowed evaluating the effectiveness of the approach. The efficiency of the approach is demonstrated using an example of solving the maximum flow problem. Multilevel structural model transformations result in the dimension reduction of a network system presentation and, consequently, a decrease in the execution time of computational procedures.

1. INTRODUCTION

Modern technical and technological objects are characterized by many elements, a variety of links and a significant amount of processed information. Complex systems are systems in which the pattern of interactions between a system's constituent parts is itself complex and is evolving together with the system's dynamics. Complex systems are those where the number of states determined by the states of elements or by the interrelations between elements is combinatorically large or uncountable. It provides the system with essential properties and imposes a number of restrictions on the study of such systems. Complex systems require specific methods for analysis and design. The main purpose of these methods is that they reduce the system to a smaller dimension, using its aggregation or decomposition.

Three groups of problems should be solved for them:

(1) The analysis of properties and behavior of the system, depending on its structure and the value of its parameters.

(2) The selection of the structure and values of the parameters based on the properties of the system.

(3) The construction of complex systems.

In the context of network theory, a complex network is a network with non-trivial topological features and with a multitude of non-trivial statistical challenges [1, 2].

The subject of research is network systems including discrete, dynamic and stochastic systems with discrete or continuous-time, where the flow balance law takes place [3].

This work is devoted to the problems of formal transformations of the structural models of network systems with the purpose of reducing the dimension of the system, when the time of evaluating system metrics plays an important role in solving real-time control problems, and when the time of modeling the system is a critical parameter.

2. LITERATURE REVIEW

Over the past decade, the number of papers devoted to the study of so-called «complex networks» or, in other words, complex systems with network topology, has increased.

Networks are used as a common model of a wide variety of complex systems. The following classification of complex networks is widely used at present [4, 5]: Technological, biological, ecological and social. Much attention is paid to the theory of networks, network modeling.

Many complex systems can be represented as multilevel networks composed by distinct elements, interacting and depending on each other. The basic elements of real-world systems are connected by different types of interactions. For example, in technological networks, a network element can be hardware, software, data, processes (including processes for providing service to users), facilities; in the case of social networks, in which the same set of people might have political or financial relationships, or might be interacting using different platforms like e-mail, Twitter, Facebook, phone calls, etc.; in ecological networks, network models are composed of a set of compartments, describing either species or functional groups, and a set of links that represent interactions or energy or biomass flows among compartments.

One of the challenges for ecological and biological-network models is to study deferent properties of systems. For example, complexity and stability, controllability and observability. To study the properties of ecological systems, multilevel network models are constructed using a series of aggregation process. The construction of ecological multilevel network models is extremely difficult problem because of the limitations in data availability associated with the difference in type of network models, level aggregation and timescale. The authors [6] state that machine learning and better data sharing between ecologists represent very important areas for advances in ecological networks.

It is known that the mathematical model of a technological network system consists of two parts: a description of its elements and a description of the structure of the system. The formal technics describing mathematical models of technological systems is well studied and includes: queueing theory, Petri nets, process algebra, and set theory.

A number of problems of the system analysis require an investigation of the structural model of a complex system. At the same time, some of them are solved only by transforming the existing structure into a kind that allows to achieve the solution of the tasks assigned to the research.

The authors [7] apply the aggregation of elements of a queuing network with the purpose of reducing its dimension and decreasing simulation time. The main point of the aggregation consists in dividing the network into two parts - the main nodes, being of interest, and the remaining nodes. It is claimed that the method is effective only in a limited range of statistical characteristics of the nodes.

The authors [8] consider the method of investigating a complex network system based on its decomposition. The method reduces the dimension of the system model and improves the performance of the modeling system on parallel platforms.

Concept of aggregation is used it the paper [9] to finding shortest paths in a graph. This is used for the satellite navigation to be able to efficiently respond in real-time to traffic updates.

The authors [10] investigate aggregation schemes for Markov processes. The approach was to lump states of a Markov process together in groups and propose a Markov process on the set of groups which has the aggregated stationary probability. The potential benefits are efficient computation, including recomputation to take into account local changes.

In the paper [11], the authors demonstrate how structure model metrics of a complex network system can be used to create random networks. The developed random networks estimate the types of network failures and their associated consequences.

The tasks of topological analysis of systems represent a range of complex problems, the solution of which requires large computing resources and the development of mathematical methods. They can be combined into such groups:

(1) Development of a valid structural description of a complex system. The tasks of this group compose a topological structure on the base of an original specification of a complex system, that is, determine elements and subsystems of a complex system and their connectivity.

(2) Determination of characteristics of a complex system with a specified topological structure. For example, a definition of strongly connected components, shortest paths, cycles, races, etc. In addition, the system structure model is used to analyze the quality metrics of the structure. For example, the number of links among subsystems; weighted number of links, whose weights are usually the functions of length, bandwidth or other characteristics of the channels, the rate of message transmission over channels, and so on.

(3) Optimal design, equivalent transformations of the topological structure of a complex system.

Most of the tasks of the third group are tasks of the increased complexity. The structure design for network systems is one of the main tasks and consists in choosing the optimal scheme for connecting nodes, selecting the transmission capacity of channels and optimal routing. The choice of a topological structure is carried out using various criteria and takes into account the constraints for time delay, the reliability of information transfer, etc.

The problem of equivalent transformations considered in this paper belongs to the third group of topological analysis tasks. Equivalent transformations can be applied in the following cases:

(1) Redistribution of links and interaction schemes within the framework of the initial structure.

(2) Aggregation and decomposition of the system components.

(3) Analysis of the model structure for detecting parallelism, deadlocks and solving the problem of mutual exclusions.

(4) Reducing the dimension of a system, when the time of metrics evaluation plays an important role in solving real-time control tasks, and also when the time of modeling the system is a critical parameter.

3. CONSTRUCTION OF THE SYSTEM STRUCTURE MODEL AND MULTILEVEL TRANSFORMATION ALGORITHM

This section is devoted to the construction of a system structure model and development of an algorithm for its multi-level aggregation. Here, we consider the problems related only to transformations of an element connection scheme. The dynamic of a system is not considered.

A complex system S contains elements C_i ($i=\overline{0,N}$), where N is a fixed number, and an external environment denoted by C_0. Let’s consider the formal model of a complex system structure [12, 13].

We formulate the first assumption as follows. Elementary signals are transmitted in a system over elementary channels. The elementary channel l connected to the output of the element C_j can transmit only elementary signals $y_l^{(j)}(t)$ having a fixed index l.

This assumption admits the following interpretation. The input of element C_j consists of m_j input contacts; the contact $X_i^{(j)}$ receives the elementary signals $x_i^{(j)}(t)$; $i=\overline{0,m_j}$. Similarly, the output of element C_j consists of r_j output contacts; the contact $Y_l^{(j)}$ gives out the elementary signals $y_l^{(j)}(t)$ which are accepted by one or more elements; $l=\overline{0,r_j}$. Thus, the mathematical model of the element C_j used for the formal description of its connection with other elements is a pair of sets: $\left[X_i^{(j)}\right]_1^m$ and $\left[Y_l^{(j)}\right]_1^r$, where for simplicity we use the notations $m = m_j$, $r = r_j$.

The second assumption. Not more than one elementary channel is connected to the input contact of any element of system; any finite number of elementary channels can be connected to the output contact.

We introduce the single-valued operator [14]:

$$Y_l^{(k)} = R\left(X_i^{(j)}\right) \tag{1}$$

where, the domain of the operator is the set $\bigcup_{j=0}^{N}\left[X_i^{(j)}\right]_1^m$ and the codomain of the operator is the set $\bigcup_{k=0}^{N}\left[Y_l^{(k)}\right]_1^r$.

The operator in Eq. (1) assigns the output contact $Y_l^{(k)}$ to the input contact $X_i^{(j)}$. If no elementary channel is connected to the contact $X_i^{(j)}$ in the system under consideration, then the operator R is not defined for this $X_i^{(j)}$. The operator in Eq. (1) we will call the operator of elements connections.

The operator of elements connections, its domain and codomain we will call the elements connections scheme of system or the formal model of the system structure. The elements connections scheme of system contains exhaustive information about the connections of system elements.

The form of the operator R plays an important role. It will affect the performance of any data processing algorithms especially in the case of large systems.

Consider the system, which structure is represented in Figure 1.

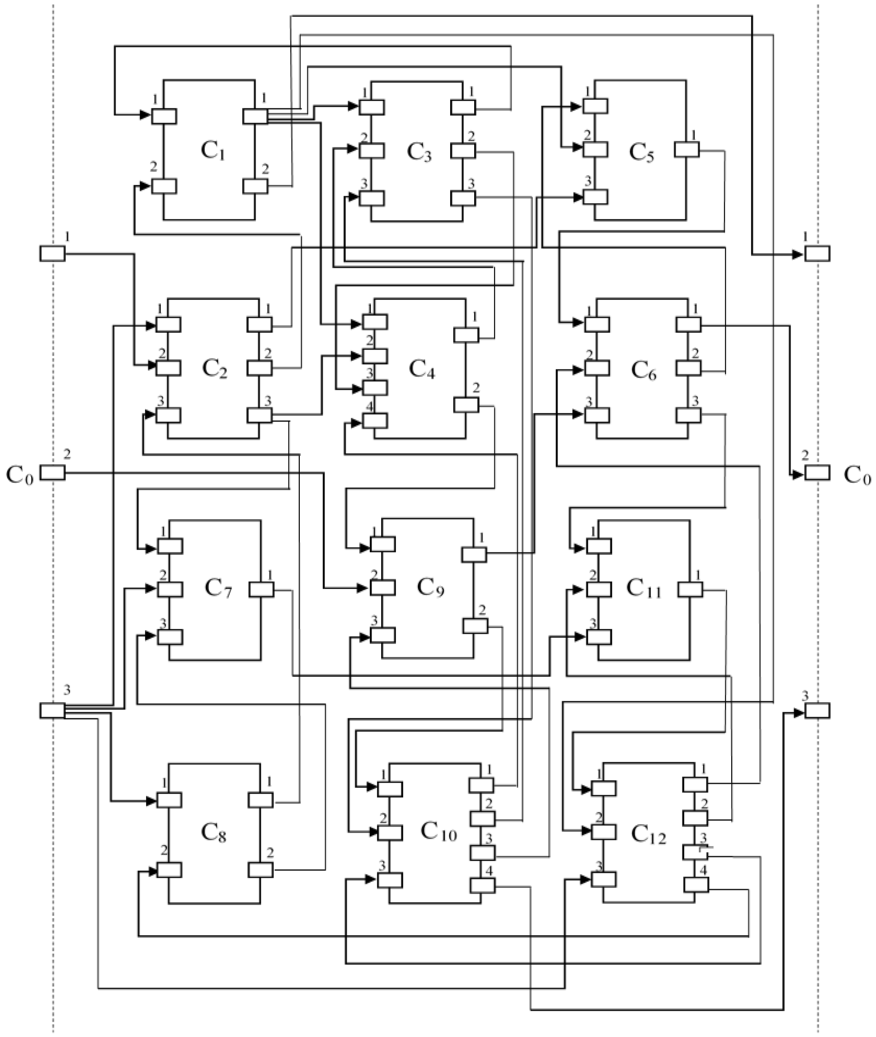

Figure 1. The structure of system S

Note, the inverse operator R_j^{-1} is not single-valued.

$$X_i^{(j)}=R^{-1}(Y_l^{(k)}) \tag{2}$$

Create the operator of elements connections R in a tabular form. The Table 1 shows the values of the operator R for the considered example of the system S.

In the Table 1, the numbers of rows correspond to the numbers of elements. The numbers of columns correspond to the numbers of the input contacts of elements. At the intersection of the row j and the column i there is a pair of numbers (k, l) indicating the number of the element k and the number of its output contact l, to which the contact $X_i^{(j)}$ is connected.

Table 1. The operator R of the elements connections for the system S

j\i	1	2	3	4
0	1,2	6,1	10,4	-,-
1	3,1	2,2	-,-	-,-
2	0,3	0,1	8,1	-,-
3	1,1	4,1	10,2	-,-
4	1,1	2,3	3,2	10,1
5	6,2	1,1	2,1	-,-
6	5,1	12,1	9,1	-,-
7	2,3	0,3	8,2	-,-
8	0,3	12,4	-,-	-,-
9	4,2	0,2	10,3	-,-
10	9,2	3,3	12,3	-,-
11	6,3	12,2	7,1	-,-
12	11,1	1,1	0,3	-,-

Let's represent the system S as an aggregate of a certain number of subsystems S_μ, containing at least one element, where $\mu = (\mu_0,\mu_1,\mu_2\dots\mu_M)$. Moreover, the element C_j, must belong to only one of the subsystems S_μ. The subsystem $S_{\mu 0}$ will include only one element C_0 representing the external environment.

The aggregation of the considered system on the subsystems can be realized as follows: $S\mu_0 = \{C_0\}$; $S\mu_1 = \{C_1, C_2\}$; $S\mu_2 = \{C_3, C_4\}$; $S\mu_3 = \{C_5, C_6\}$; $S\mu_4 = \{C_7, C_8\}$; $S\mu_5 = \{C_9, C_{10}\}$; $S\mu_6 = \{C_{11}, C_{12}\}$. The aggregation proposed is depicted in the Figure 2.

It is obvious that a subsystem $S_{\mu i}$, on the one hand, can itself be an independent system, just like the system S, and on the other hand, it can be an element of the system S. Further, for simplicity the subsystem $S_{\mu i}$ will be considered as S_μ.

In this paper, the construction of the elements' connections scheme and its implementation for these two cases are considered. In any case, the construction of the elements connections scheme consists of two steps: definition of the fictitious contacts on the border of the subsystem $S_{\mu i}$ and construction of the operator of the elements connections.

Step 1. Definition of the fictitious contacts. Each subsystem on the border must have fictitious input and fictitious output contacts for communication with other subsystems of the system S. The fictitious contacts play the role of male-female connectors that connect the blocks of complex electronic devices.

In the first case, the subsystem S_μ, considered as an independent system, has to have an external environment, which is denoted as an external element C_0^μ or subsystem $S'_{\mu 0}$.

The internal structure of the external environment $S'_{\mu 0}$ is invisible to the elements of the subsystem S_μ. The subsystem S_μ interacts with the external environment $S'_{\mu 0}$ through its fictitious input contacts $X_i^{(0)\mu}$, which are connected to output contacts of the elements of the subsystem S_μ, and through its fictitious output contacts $Y_l^{(0)\mu}$, which are connected to input contacts of the elements of the subsystem S_μ.

In the second case, the system S is divided into several subsystems S_μ. Each of the subsystems is considered as an element of the system S. The internal structure of a subsystem S_μ is invisible to other subsystems of the system S. Each subsystem (on the border) must also have fictitious input $X_i^{(\mu)}$ and fictitious output $Y_l^{(\mu)}$ contacts for communication with other subsystems of the system S.

Combining these two cases, we come to the fact that the pair of contacts $X_i^{(\mu)}$ and $Y_j^{(0)\mu}$ and also $X_i^{(0)\mu}$ and $Y_l^{(\mu)}$ are combined into *double* fictitious contacts on the border of the subsystem S_μ (Figure 2).

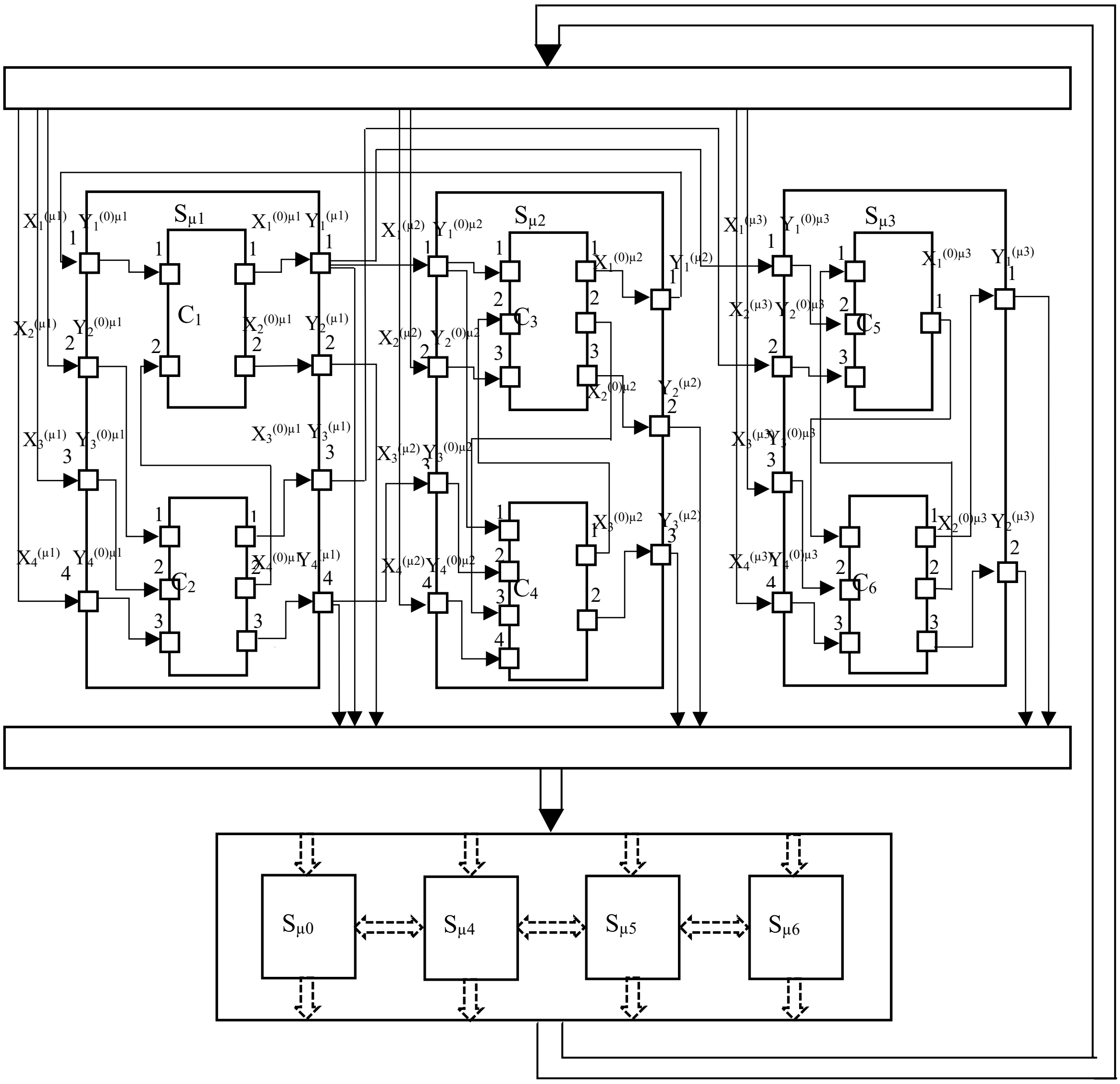

Figure 2. The aggregation of the system S on the subsystems $S_{\mu i}$

Introducing the concept of fictitious contacts for each subsystem S_μ plays an important role. In a project, if necessary, it is easy to replace the subsystem S_μ with a new one. In comparison with the old subsystem S_μ, a new one can: use a faster circuitry, which can improve the performance of a system S; allow extending functionality, for example, by integrating additional security mechanism and so on.

Determine the fictitious contacts. It is obvious that the input and output fictitious contacts of the subsystem S_μ are defined for the elements of two sets:

(1) $[Y_l^{(j)}]_\mu$ is a set of the output contacts of all elements C_j, where $C_j \in S_\mu$, connected to the input contacts of the elements C_k, where $C_k \notin S_\mu$, as well as to the input contacts of the external environment C_0.

(2) $[X_i^{(j)}]_\mu$ is a set of the input contacts of all elements C_j, where $C_j \in S_\mu$,, connected to the output contacts of the elements C_k, where $C_k \notin S_\mu$, as well as to the output contacts of the external environment C_0.

So, the set $[Y_l^{(j)}]_\mu$ is determined as:

$$[Y_l^{(j)}]_\mu = \bigcup_{C_j \in S_\mu} \{[Y^{(j,0)}] \cup (\bigcup_{C_k \notin S_\mu} [Y^{(j,k)}])\} \tag{3}$$

where, $[Y^{(j,k)}]$ is a set of output contacts of an element C_j connected to corresponding input contacts of an element C_k, where $C_k \notin S_\mu$.

Selecting contacts $Y_l^{(j)}$ for inclusion in the set $[Y_l^{(j)}]_\mu$, in general case, we can meet the same contact $Y_l^{(j)}$ several times. According to the Idempotent Law of union of sets, the same contacts $Y_l^{(j)}$ are not repeated. Thus, for each $Y_l^{(j)} \in [Y_l^{(j)}]_\mu$ it is sufficient to have only one fictitious contact $Y_l^{(\mu)}$.

Each $Y_l^{(j)} \in [Y_l^{(j)}]_\mu$ generates a *double* fictitious contact on the border of the subsystem S_μ. A pair of operators is used to define this *double* fictitious contact:

$$\begin{aligned} &Y_l^{(\mu)} = Q_\mu (Y_l^{(j)}), \\ &X_i^{(0)\mu} = Q'_\mu (Y_l^{(j)}) \end{aligned} \tag{4}$$

The operators in Eq. (4) are called the numbering operators of fictitious contacts of the subsystem S_μ. The operator Q_μ defines an output fictitious contacts $Y_l^{(\mu)}$ of the subsystem S_μ (second case) (Figure 2). The operator Q'_μ defines an input fictitious contacts $X_i^{(0)\mu}$ of the external environment $S'_{\mu 0}$ (first case). In fact, each of these operators symbolizes a procedure of assigning values to fictitious contacts. The values of the operators in Eq. (4) can be represented by a table of numbering fictitious contacts.

As an example, the Table 2 contains the numbers of fictitious contacts of the subsystem $S_{\mu 1}$, connected to contacts of the set $[Y_l^{(j)}]_{\mu 1}$ with the help of operators Q_μ and Q'_μ.

Table 2. The values of the operators Q_μ and Q'_μ

$Y_l^{(j)}$ (j,l)	1,1	1,2	2,1	2,3
Qμ	1	2	3	4
Q′μ	1	2	3	4

Similarly, we consider the formation of the set $[X_l^{(j)}]_\mu$. We use the expression:

$$[X_l^{(j)}]_\mu = \bigcup_{C_j \in S_\mu} \{[X^{(j,0)}] \cup (\bigcup_{C_k \notin S_\mu} [X^{(j,k)}])\} \tag{5}$$

where, $[X^{(j,k)}]$ is the set of input contacts of an element C_j connected to the corresponding output contacts of an element C_k, where $C_k \notin S_\mu$.

According to the Idempotent Law of union of sets, only different contacts $X_l^{(j)}$ enter the set $[X_l^{(j)}]_\mu$. For numbering of the fictitious contacts $X_i^{(\mu)}$, we introduce the pair of operators:

$$\begin{aligned} &X_i^{(\mu)} = P_\mu (X_i^{(j)}), \\ &Y_l^{(0)\mu} = P'_\mu (X_i^{(j)}) \end{aligned} \tag{6}$$

The operators in Eq. (6) are called the numbering operators of the fictitious contacts of the subsystem S_μ. The operator P_μ defines the input fictitious contacts $X_i^{(\mu)}$ of the subsystem S_μ (second case) (Figure 2). The operator P'_μ defines the output fictitious contacts $Y_l^{(0)\mu}$ of the external environment $S'_{\mu 0}$ (first case). In fact, each of these operators symbolizes a procedure of assigning values to the fictitious contacts. The values of the operators in Eq. (6) can be represented by the table of numbering the fictitious contacts.

Table 3. The values of the operators P_μ and P'_μ

$X_i^{(j)}$ (j,i)	1,1	2,1	2,2	2,3
Pμ	1	2	3	4
P′μ	1	2	3	4

As an example, the Table 3 contains the numbers of fictitious contacts of the subsystem $S_{\mu 1}$, connected to contacts of the set $[X_l^{(j)}]_{\mu 1}$ with the help of operators P_μ and P'_μ.

Step 2. Construction of the operator of the elements connections. This step, in turn, should be considered for two cases: first case, a subsystem S_μ is considered as an independent system, just like the system S; second case, a subsystem S_μ is considered as an element of the system S.

Construction of the operator of the elements connections for first case. Denote this operator by R_μ.

Consider a subsystem Sμ as an independent system. It means that all other elements of the system S including C_0 represent the external environment $S'_{\mu 0}$ in relation to the subsystem S_μ. There are two types of connections within the elements of the subsystem S_μ.

The connections of first type are internal connections of input and output contacts of elements C_j, where $C_j \in S_\mu$.

The connections of second type are connections of input and output contacts of the elements C_j of the subsystem S_μ to input and output contacts of the element $S'_{\mu 0}$. The external environment $S'_{\mu 0}$ interacts with the subsystem S_μ through its input contacts $X_i^{(0)\mu}$, which are connected to output contacts of elements C_j, where $C_j \in S_\mu$, and through the output contacts $Y_l^{(0)\mu}$, which are connected to input contacts of elements C_j, where $C_j \in S_\mu$.

In the first case, the sets of the contacts $[X_i^{(j)}]_1^m$ and $[Y_l^{(j)}]_1^r$ of the elements C_j, where $C_j \in S_\mu$ are given. The operator of the elements connections R_μ for this case is equal to R.

In the second case, the subsystem S_μ has to connect with the subsystem $S'_{\mu 0}$ in the following manner: input contacts of the subsystem $S'_{\mu 0}$ to output contacts of the elements of the subsystem S_μ and output contacts of the subsystem $S'_{\mu 0}$ to input contacts of the elements of the subsystem S_μ. Thus, the operators Q'_μ in Eq. (4) and P'_μ in Eq. (6) define the input $X_i^{(0)\mu}$ and output $Y_l^{(0)\mu}$ contacts respectively:

$$\begin{aligned} X_i^{(0)\mu} &= Q'_\mu(Y_l^{(j)}), \\ Y_l^{(0)\mu} &= P'_\mu(X_i^{(j)}) \end{aligned} \tag{7}$$

Finally, the operator R_μ is determined as:

$$Y_l^{(k)} = R_\mu\left(X_i^{(j)}\right) \tag{8}$$

where, the domain of the operator is the set:

$$\left\{\left[X_i^{(0)\mu}\right]_l^m \cup \left(\bigcup_{C_j \in S_\mu}\left[X_i^{(j)}\right]_l^m\right)\right\} \tag{9}$$

and the codomain of the operator is the set:

$$\left\{\left[Y_l^{(0)\mu}\right]_l^r \cup \left(\bigcup_{C_j \in S_\mu}\left[Y_l^{(k)}\right]_l^r\right)\right\} \tag{10}$$

Consider an algorithm of forming a set of codomains of the operator R_μ. Analyzing the expression in Eq. (9), we conclude that the domain of R_μ consists of three subsets.

The first subset. A contact $X_l^{(j)}$ of an element C_j, where $C_j \in S_\mu$, is connected to a contact $Y_l^{(k)}$ of an element C_k, where $C_k \in S_\mu$. It can be written:

$$[X_l^{(j)}]_\mu = \bigcup_{C_k \in S_\mu}[X^{(j,k)}] \tag{11}$$

In this particular case, it is evident the contact $Y_l^{(k)}$ is defined as

$$Y_l^{(k)} = R(X_i^{(j)}) \tag{12}$$

Thus, the operator R_μ is equal to the operator R:

$$R_\mu = R \tag{13}$$

A procedure of forming the first subset in Eq. (11) of the subsystem S_μ consists of two steps.

Step 1. Consider all rows j (where $C_j \in S_\mu$) and the pairs of numbers (k, l) in the Table 1. If the first number k is such that $C_k \in S_\mu$, then the second one l denotes the number of the output contact of the element C_k connected to the input contact $X_i^{(j)}$.

Step 2. The value of the operator R_μ has to be determined, using the expression in Eq. (11).

The two-step operation has to be executed for all rows j of the Table 1, where $C_j \in S_\mu$.

The second subset. A contact $X_i^{(j)}$ of the element C_j, where $C_j \in S_\mu$, is connected to the contact $Y_l^{(k)}$ of the element C_k, where $C_k \notin S_\mu$. It can be written:

$$[X_l^{(j)}]_\mu = \bigcup_{C_k \notin S_\mu}[X^{(j,k)}] \tag{14}$$

In this case, it is necessary to form a fictitious contact $Y_l^{(0)\mu}$ on the border of the subsystem S_μ, using the operator P'_μ in Eq. (7). The procedure of forming the second subset in Eq. (14) of the subsystem S_μ consists of two steps.

Step 1. Consider all rows j (such as $C_j \in S_\mu$) and the pairs of the numbers (k, l) in the Table 1. If the first number k is such that $C_k \notin S_\mu$, then the second number l is the number of the output contact of the element C_k connected to the input contact $X_i^{(j)}$.

Step 2. The operator R_μ has to be determined. It is equal to the operator P'_μ in Eq. (7). So,

$$R_\mu = P'_\mu \tag{15}$$

Finally, the fictitious contact has to be found as:

$$Y_l^{(0)\mu} = R_\mu(X_i^{(j)}) \tag{16}$$

The third subset. Consider the third subset that consists of fictitious input contacts $X_i^{(0)\mu}$ of the external environment $S'_{\mu 0}$. The operator Q'_μ in Eq. (7) has to be used in order to determine a contacts $Y_l^{(k)}$, which is connected to the fictitious contacts $X_i^{(0)\mu}$. It is easy to conclude that:

$$Y_l^{(k)} = (Q'_\mu)^{-1}(X_i^{(0)\mu}) \tag{17}$$

where, the operator $(Q'_\mu)^{-1}$ is an inverse operator Q'_μ.

Thus,

$$R_\mu = (Q'_\mu)^{-1} \tag{18}$$

Note, the operator Q'_μ is a one-to-one operator.

Finally, the procedure of constructing the operator R_μ is defined by the expression:

$$Y_l^{(k)} = \begin{cases} R\left(X_i^{(j)}\right) \text{for } X_i^{(j)} \in \bigcup\limits_{C_j \in S_\mu}\bigcup\limits_{C_k \in S_\mu}\left[X^{(j,k)}\right] \\ P_\mu^1\left(X_i^{(j)}\right) \text{for } X_i^{(j)} \in \bigcup\limits_{C_j \in S_\mu}\bigcup\limits_{C_k \notin S_\mu}\left[X^{(j,k)}\right] \\ (Q_\mu^1)^{-1}\left(X_i^{(0)\mu}\right) \text{for } X_i^{(0)\mu} \in \left[X_i^{(0)\mu}\right]_l^m \end{cases} \tag{19}$$

As an example, the values of the operator R_μ where $\mu = \mu 1$ are given in Table 4. Like the Table 1, the Table 4 shows the values of the operator $R_{\mu 1}$. The row 0 in this table corresponds to the external environment $S'_{\mu 0}$. The fictitious contacts $X_i^{(0)\mu}$ and $Y_l^{(0)\mu}$ correspond to the input and output contacts of $S'_{\mu 0}$, respectively. The rows 1 and 2 correspond to the elements C_1 and C_2, respectively. The number of columns of this table is equal to the maximum number of input contacts among the C_1, C_2 and $S'_{\mu 0}$ and is equal to 4. At the intersection of rows and columns there are pairs of numbers (k, l) indicating the number of the element k (where $C_k \in S_\mu$ and $S'_{\mu 0}$) and the number of its output contact l, to which the input contact $X_i^{(j)}$ is connected.

The case, where a subsystem S_μ is considered as an independent system has been applied to the design of secure hardware systems. In the paper [15], hardware obfuscation technique on the base of proposed formalism is described.

Table 4. The operator R_μ for the subsystem $S_{\mu 1}$

j\i	1	2	3	4
0	1,1	1,2	2,1	2,3
1	0,1	2,2	-,-	-,-
2	0,2	0,3	0,4	-,-

Now, consider construction of the operator of the elements connections for the case where a subsystem S_μ is considered as an element of the system S (second case). In fact, that is an operator of subsystems connections or a two-level operator of the elements connections. Denote this operator by R_{II}. Similar to the operator R, determine the operator R_{II} as:

$$Y_l^{(v)} = R_{II}(X_i^{(\mu)}) \tag{20}$$

where, the domain of the operator is the set of fictitious output contacts of subsystem S_μ $\bigcup_{\mu=\mu_0}^{\mu_M} [X_i^{(\mu)}]_1^m$, and the codomain of the operator is the set of fictitious input contacts of subsystem S_μ $\bigcup_{\mu=\mu_0}^{\mu_M} [Y_i^{(\mu)}]_1^r$.

If no elementary channel is connected to the contact $X_i^{(\mu)}$, then the operator R_{II} is not defined for this $X_i^{(\mu)}$.

Consider an algorithm of forming a set of codomains of the operator R_{II}. A procedure of the operator R_{II} is based on an analysis of a chain to which the contacts $X_i^{(\mu)}$ and $Y_l^{(v)}$ belong. This procedure consists of two steps.

Step 1. For a fictitious contact $X_i^{(\mu)}$, there is always a contact $X_i^{(j)}$ (where $C_j \in S_\mu$), which can be determined using the expression:

$$X_i^{(j)} = P_\mu^{-1}(X_i^{(\mu)}) \tag{21}$$

where, P_μ^{-1} is the inverse operator P_μ in Eq. (6).

The operator $P_\mu^{-1}(X_i^{(\mu)})$ is not a one-valued operator. Thus, there can be more than one contact $X_i^{(j)}$ for a fictitious contact $X_i^{(\mu)}$, but it does not matter. If the fictitious contact $X_i^{(\mu)}$ corresponds to more than one contact $X_i^{(j)}$, the output contact $Y_l^{(k)}$, where $C_k \notin S_\mu$, will always be the same.

Step 2. Definition of a fictitious output contact $Y_l^{(v)}$. If for a contact $X_i^{(j)}$ there exists an output contact $Y_l^{(k)}$ of the component C_k, such that $C_k \notin S_\mu$ and $C_k \in S_v$, then there always exists an operator:

$$Y_l^{(k)} = R(X_i^{(j)}) \tag{22}$$

Therefore, we define the required fictitious contact $Y_l^{(v)}$ with the help of the operator in Eq. (4):

$$Y_l^{(v)} = Q_v(Y_l^{(k)}) \tag{23}$$

Thus,

$$R_{II} = Q_v \tag{24}$$

Consider a particular case, when k C_0, fictitious contacts $Y_l^{(\mu 0)}$ and $X_i^{(\mu 0)}$ of the subsystem $S_{\mu 0}$ coincide with the corresponding contacts $Y_l^{(0)}$ and $X_i^{(0)}$ of the element C_0. Formally, the contacts $Y_l^{(\mu 0)}$ and $X_i^{(\mu 0)}$ are obtained as a result of applying the operators $Q_{\mu 0}$ in Eq. (4) and $P_{\mu 0}$ in Eq. (6), respectively. In this case, the value of the operator R_{II} is obtained using the operator:

$$Y_l^{(\mu 0)} = Q_{\mu 0}(Y_l^{(0)}) \tag{25}$$

Finally, the operator R_{II} is defined by the expression:

$$Y_l^{(\nu)} = \begin{cases} Q_\nu(Y_l^{(k)}), \text{if } k \neq 0, C_k \in S_\nu \\ Y_l^{(?\ )} = Q_{\mu 0}\ (Y_l^{(0)}), \text{if } k = 0 \end{cases} \tag{26}$$

where, $Y_l^{(k)} = R[P_\mu^{-1}(X_i^{(\mu)})]$.

The values of the operator R_{II} for the considered example are given in Table 5.

Table 5. The operator R_{II} for the considered system S

j\i	1	2	3	4
0	1,2	3,1	5,4	-,-
1	2,1	0,3	0,1	4,2
2	1,1	5,3	1,4	5,2
3	1,1	1,3	6,1	5,1
4	1,4	0,3	6,3	-,-
5	2,3	0,2	2,2	6,2
6	3,2	4,1	1,1	0,3

Like the Table 1, the Table 5 shows the values of the operator R_{II}. The row number in the table corresponds to the subsystem numbers of the second level. The column number in the table correspond to the number of fictitious input contact of subsystem of the second level. At the intersection of the rows and the columns there are pairs of numbers (k, l) indicating the number of the subsystem k and the number of its fictitious output contact l, to which fictitious input contacts of subsystems are connected.

To solve many practical problems related to network systems, it is necessary to aggregate subsystems S_μ into larger subsystems S^3_μ (third level), and those, in turn, into even larger ones, etc. In this case, it is necessary to consider a three-level R_{III} or four-level operator of the elements connections R_{IV}, and so on. Algorithmic implementation of operator construction for higher levels is invariant. Like the second level the construction of the subsystem's connections scheme for third and larger level consists of two steps: definition of the fictitious contacts on the border of the subsystem S^3_μ; construction of the operator of the elements connections. The content of each of these steps for each level is the same. The construction of a multilevel aggregation scheme can be performed recurrently.

Table 6. The operator R_{III} for the considered example of the system S

j\i	1	2	3	4	5
0	1,2	3,1	2,3	-,-	-,-
1	2,1	0,3	0,1	3,3	-,-
2	1,1	1,4	0,2	3,2	-,-
3	1,1	1,3	2,2	1,5	0,3

As an example, in the work the third level aggregation was performed. The subsystems S_μ of second level have been aggregated as follows: $S^3\mu_0 = \{C_0\}$; $S^3\mu_1 = \{S\mu_1, S\mu_4\}$; $S^3\mu_2 =$

$\{S\mu_2, S\mu_5\}$; $S^3\mu_3 = \{S\mu_3, S\mu6\}$. Table 6 shows the values of the operator R_{III} for the considered example of the system S.

Thus, the application of the multilevel aggregation algorithm shows that, in comparison with the first level, at the third level of the system the number of elements has decreased by sixty-nine per cent and the number of connections has decreased by fifty-eight per cent.

It is easy to show that all elements connections schemes (one-level, two-level and three-level and so on) are equivalent from the following point of view. To each elementary channel connecting the contacts $X_i^{(j)}$ and $Y_l^{(k)} = R(X_i^{(j)})$ in one-level elements connections scheme corresponds an elementary channel connecting these contacts in multilevel elements connections schemes.

4. PROGRAM IMPLEMENTATION

This section is devoted to the description of the program implementation. The algorithm of multilevel aggregation of the network system has been programed. The program is written in the Java programming language and consists of fourteen classes, ten of which display the structure and functionality of the program, four are test classes. Simplified UML diagram with class names and fields is represented in Figure 3.

For the practical implementation of this task, it was necessary to construct classes that would reflect the structure and behavior of the system. The following structural parts were identified: a system that contains components; component; component port; connection between components. On the basis of this division such classes as SystemOfComponents, Component, Gate, Connection were constructed.

This system is a multilevel system, and to represent its levels the LevelOfSystem class was created. To store the layers of the system a collection was created in the SystemOfComponents class.

To create a hierarchical structure, the components were combined into subsystems. The component is both a component and an independent system. For the software implementation of this aspect, the inheritance mechanism was applied and, as consequence, the Component class was inherited from the LevelOfSystem class.

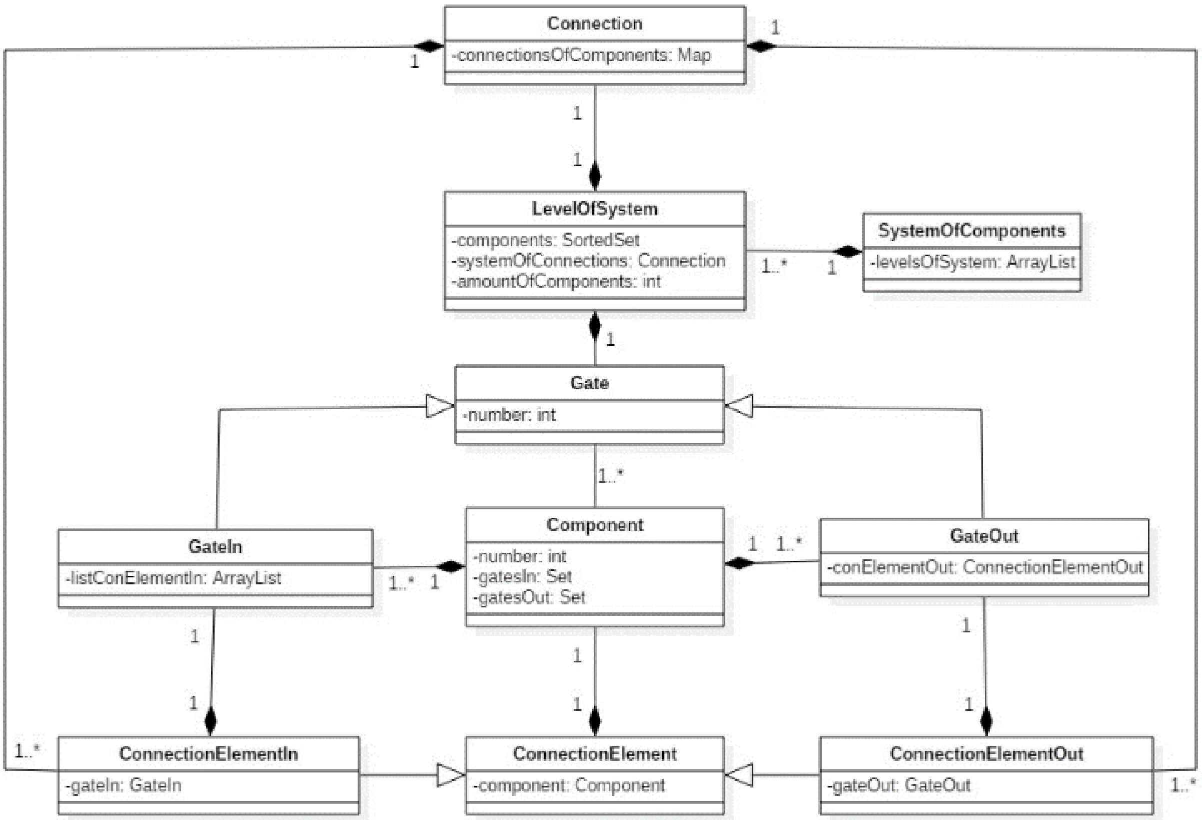

Figure 3. Simplified UML diagram with class names and fields

Each component has input and output ports which have a fundamental difference: only one connection can enter the input port, while several connections may leave the output port. However, at the same time, these ports have the same functions. Thus, to represent the ports of the component, the Gate class was created.

The next task was to represent the connection between two components. To solve it, the Connection class was created. To determine the beginning and the end of communication it was necessary to create a certain object that would store the component and port from which the connection leaves, and the component and port into which this connection enters. Thus, the following classes were created: the ConnectionElement class that stores the component and the ConnectionElementIn and ConnectionElementOut classes that inherit ConnectionElement class and store the input and output ports respectively. To create a connection between components, a mapping collection was created in the Connection class, the key of which is the ConnectionElementIn object, and the value is ConnectionElementOut.

As a result of the program implementation, the Tables 46 were obtained, representing the connections of the subsystems at each level of the system.

5. EFFECTIVENESS ASSESSMENT OF MULTILEVEL TRANSFORMATIONS OF SYSTEM STRUCTURE MODEL

The formal transformations of the system structure model, using a multilevel aggregation, results in reduction in computational complexity and faster modeling. The efficiency of the approach is demonstrated using an example of solving the maximum flow problem [16]. The maximum flow problem belongs to the group of topological analysis problems whose purpose is to distribute network flows to achieve the maximum values of communication efficiency.

The maximum flow problem is formulated as follows: the maximum possible total value of the flow between the source and the sink has to be found for given network with established initial distribution of flows for graph edges and capacities. It means that the flow has to be increased if it has not reached the maximum value. The maximum flow value is equal to the sum of weights of the edges in the minimum cut in accordance with the theorem proved by Ford and Fulkerson, which is applied for solving the maximum flow problem [17].

The Ford-Fulkerson algorithm consists of the following steps.

(1) A path from the souse to the sink which is called an augmenting path has to be found in the given flow network.

(2) The maximum value of the flow in the augmenting path which is called the residual capacity has to be found. Each edge in the augmenting path must be labeled with the its capacity and residual capacity via slash notation. The flow value must be nonnegative and must not exceed the given capacity, but can be equal to it.

(3) If the flow is equal to the capacity of the edge, this edge is saturated. As a result, it cannot be considered in searching next augmenting path.

(4) Enumeration finishes when transfer from the source to the sink becomes impossible due to no augmenting paths exist.

(5) The value of the maximum flow is equal to the capacity of the minimum cut. It can be found as the sum of the flow values of the edges which are incoming to the sink.

The maximum flow of the network is calculated according to the formula:

$$f = \sum_{i=1}^{n} f_i \tag{27}$$

where, n is the amount of the edges which are incoming to the sink; f_i is the flow value of the edge which is incoming to the sink.

The average length of the augmenting path is calculated according to the formula:

$$L_{avg} = \frac{\sum_{i=1}^{n} L_i}{n} \tag{28}$$

where, n is the amount of the augmenting paths; L_i is the length of the augmenting path.

Implementation of the approach is shown below. To evaluate the efficiency of the approach proposed, an experiment has been performed. For each of three levels of structural system model the maximum flow has been calculated. Consider this experiment in detail.

First, find the maximum flow for the initial network (first level) applying the Ford-Fulkerson algorithm. For the initial network (Figure 1), we assign the capacity values of the channels, which connect the components. We transform the Table 1 of the elements connections into the Table 7, where the channel capacities are indicated by the third values.

The graph of the system was constructed (Figure 4), where the source combines the output contacts of the external environment and the sink combines the input contacts of the external environment. The weight of each edge is equal to the capacity of the corresponding channel.

The search of the maximum flow was fulfilled using the Ford-Fulkerson algorithm. An augmenting path from the source to the sink was chosen on each iteration. Each edge was labeled with capacity and maximum possible flow via slash notation. The enumeration of the possible augmenting paths finished when transfer from the source to the sink became impossible in consequence of the saturation of the edges of the graph. After that, the maximum flow of the network was found as a result of the summation of the flow values of the edges which are incoming to the sink.

Table 7. The first level elements connections with capacity values

i\j	1	2	3	4
0	1,2,20	6,1,25	10,4,10	-,-
1	3,1,15	2,2,10	-,-	-,-
2	0,3,30	0,1,75	8,1,15	-,-
3	1,1,15	4,1,15	10,2,10	-,-
4	1,1,10	2,3,5	3,2,25	10,1,10
5	6,2,15	1,1,5	2,1,5	-,-
6	5,1,15	12,1,5	9,1,25	-,-
7	2,3,15	0,3,45	8,2,25	-,-
8	0,3,60	12,4,10	-,-	-,-
9	4,2,5	0,2,45	10,3,15	-,-
10	9,2,15	3,3,10	12,3,15	-,-
11	6,3,5	12,2,10	7,1,10	-,-
12	11,1,10	1,1,10	0,3,50	-,-

The augmenting paths, the residual capacities and the number of the traversed edges are shown in the Table 8.

Table 8. Result of the graph traversal

№	Augmenting path	Residual capacity	Amount of the traversed edges
1	Source–2–1–Sink	10	3
2	Source–2–5–6–Sink	5	4
3	Source–2–4–3–1–Sink	5	5
4	Source–9–6–Sink	20	3
5	Source–9–6–11–12–10–Sink	5	6
6	Source–9–10–Sink	5	3
7	Source–9–10–3–1–Sink	5	5

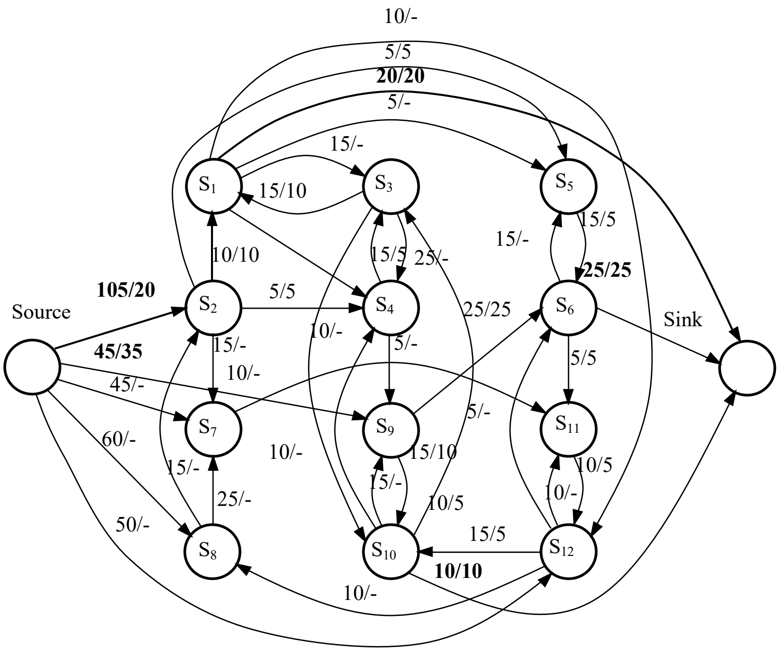

Figure 4. Graph of the first level of the system

Thus, the maximum flow of the network was found according to the formula in Eq. (27):

$$f_1 = 20 + 25 + 10 = 55$$

The seven augmenting paths were traversed during algorithm implementation. The average length of the path was found according to the formula in Eq. (28):

$$L_{avg1} = \frac{3+4+5+3+6+3+5}{7} = 4{,}14$$

By analogy with the first level, the maximum flow was found for the second and third levels.

As a result of using this method, the number of the elements and the number of the connections between the elements have decreased. Moreover, the number of graph traversal iterations has declined by fifty-seven per cent in finding the maximum network flow at the third level of the system. In addition to this, the average length of the augmenting path from the source to the sink has decreased by thirty-six per cent.

The final results of the computational experiment are given in the Table 9.

Summarizing:

(1) The value of the maximum flow for each of three levels of system structural model is the same. This is an indirect confirmation of the equivalence of multilevel structural model transformations, as well as the correctness of algorithmic and software implementation.

(2) Multilevel structural model transformations result in the dimension reduction of a network system presentation and, consequently, a decrease in the execution time of computational procedures. The results of the evaluation of the efficacy of the composition method: the number of the element of the system decreased by 69%; the number of the connections among the elements of the system decreased by 58%; the number of the graph traversal iterations decreased by 57%; average length of the path from the source to the sink decreased by 36%.

Table 9. Results of the evaluation of the effectiveness of the composition method

Parameters	First level	Second level	Third level	Result: decreased by
The number of the element of the system	13	7	4	69%
The number of the connections between elements of the system	38	26	16	58%
The number of the graph traversal iterations	7	4	3	57%
Average length of the path from the source to the sink	4,14	3	2,67	36%
Maximum flow	55	55	55	-

6. CONCLUSIONS AND FUTURE WORK

In this paper, we examined the technique of formal transformations of structural models of complex systems and its application. We demonstrated, that:

(1) Multilevel structural model transformation techniques provide a simpler representation of network systems while preserving the topological properties of the higher dimension system.

(2) The results of the work can be used to study and solve problems associated with streaming processes in networks, when throughput or other parameter changes over time and the time of system modeling is a critical parameter.

(3) The formal transformations of the structural models of complex systems with the purpose of reducing the dimension of the system, can be leveraged in the multi-tier telecommunication network management model. This will allow the management system to make decisions in real time to identify bottlenecks in a network, meet customer needs for

rapid deployment of new services, meet strict quality of service requirements.

(4) The application of the object-oriented concept in the program implementation allows making the transformation techniques invariant to the way of a formal presentation of a system structural model.

Within the framework of the future study, we can present such ideas:

(1) Development of a generator of structural models with specified topological characteristics of large-scale network systems. This will allow evaluating the effectiveness of proposed technologies to solve various optimization problems associated with large networks.

(2) In various problems of managing complex network systems, time is a critical parameter. These problems include detecting bottlenecks, parallelism, as well as mutual exclusion, deadlocks etc. Therefore, the study of the application of equivalent multilevel topological transformations for these problems is an important task.

REFERENCES

[1] Kolaczyk, E.D. (2009). Statistical analysis of network data: Methods and Models. Springer, New York. https://doi.org/10.1007/978-0-387-88146-1

[2] Newman, M. (2010). Networks: An Introduction. Oxford University Press. https://doi.org/10.1093/acprof:oso/9780199206650.001.0001

[3] Denning, P.J., Buzen, J.P. (1978). The operational analysis of queuing network models. Computing Surveys, 10(3): 225-261. https://doi.org/10.1145/356733.356735

[4] Lu, J., Chen, G., Ogorzalek, M., Trajkovic, L. (2013). Theory and applications of complex networks: Advances and challenges. Proceedings of the IEEE International Symposium on Circuits and Systems, Beijing, China, pp. 2291-2294. https://doi.org/10.1109/ISCAS.2013.6572335

[5] Landi, P., Minoarivelo, O.H., Brännström, Å., Hui, C., Dieckmann, U. (2018). Complexity and stability of ecological networks: A review of the theory. Population Ecology, 60(4): 319-345. https://doi.org/10.1007/s10144-018-0628-3

[6] Bohan, D., Dumbrell, A., Woodward, G., Jackson, M. (2018). Next Generation Biomonitoring: Part 1. Advances in Ecological Research, Academic Press.

[7] Shortle, J.F., Mark, B.L., Gross, D. (2009). Reduction of closed queueing networks for efficient simulation. ACM Transactions on Modeling and Computer Simulation, 19(3). https://doi.org/10.1145/1540530.1540531

[8] Liu, J.X., Vorst, N.V. (2012). Realizing large-scale interactive network simulation via model splitting. Proceedings of the 26th Workshop on Principles of Advanced and Distributed Simulation (PADS 2012), Zhangjiajie, China, pp. 3-12.

[9] MacKay, R.S. (2011). Hierarchical aggregation of complex systems. Proceedings of the ECCS'11, Vienna, Austria.

[10] MacKay, R.S., Robinson, J.D. (2018). Aggregation of Markov fows I: Theory. Philosophical Transactions of The Royal Society A Mathematical Physical and Engineering Sciences. https://doi.org/10.1098/rsta.2017.0232

[11] Hackl, J., Adey, B.T. (2017). Generation of Spatially Embedded Random Networks to Model Complex Transportation Networks. In: Caspeele R., Taerwe L., Proske D. (eds) 14th International Probabilistic Workshop. Springer, Cham, pp. 217-230. https://doi.org/10.1007/978-3-319-47886-9_15

[12] Mesarović, M., Mako, D., Takahara, Y. (1977). Theory of Hierarchical Multilevel Systems. New York: Academic.

[13] Grossman, P. (1995). Discrete Mathematics for Computing. Palgrave, London. https://doi.org/10.1007/978-1-349-13908-8

[14] Gorbachov, V., Batiaa, A.K., Ponomarenko, O., Romanenkov, Y. (2018). Formal transformations of stuctural models of complex network systems. Proceedings of the IEEE 9th International Conference on Dependable Systems, Services and Technologies DESSERT'2018, Kyiv, Ukraine, pp. 473-477.

[15] Gorbachov, V., Batiaa, A.K., Ponomarenko, O., Kulak, E. (2018). Securing computer hardware on the base of reference monitor obfuscation. Proceedings of the IEEE International Scientific-Practical Conference Problems of Infocommunications. Science and Technology (PIC S&T'2018), Kharkiv, Ukraine, pp. 406-410.

[16] Cormen, T., Leiserson, C., Rivest, R., Stein, C. (2009). Introduction to Algorithms, Third Edition, The MIT Press.

[17] Ford L.R., Fulkerson D.R. (2009) Maximal Flow Through a Network. In: Gessel I., Rota GC. (eds) Classic Papers in Combinatorics. Modern Birkhäuser Classics. Birkhäuser Boston. https://doi.org/10.1007/978-0-8176-4842-8_15