UDC 004.02 Doi: 10.31772/2712-8970-2023-24-3-418-435

Для цитирования: Гаипов К. Э., Кузнецов А. А., Крикунов И. Л. Применение контурного метода для решения задачи оптимального распределения трафика в сети // Сибирский аэрокосмический журнал. 2023. Т. 24, № 3. С. 418–435. Doi: 10.31772/2712-8970-2023-24-3-418-435.

For citation: Gaipov K. E., Kuznetsov A. A., Krikunov I. L. [Using of the contour method to solving the problem of optimal traffic distribution in the network]. *Siberian Aerospace Journal*. 2023, Vol. 24, No. 3, P. 418–435. Doi: 10.31772/2712-8970-2023-24-3-418-435.

Применение контурного метода для решения задачи оптимального распределения трафика в сети

К. Э. Гаипов, А. А. Кузнецов, И. Л. Крикунов

Сибирский государственный университет науки и технологий имени академика М. Ф. Решетнева Российская Федерация, 660037, г. Красноярск, просп. им. газ. «Красноярский Рабочий», 31 E-mail: gaipovke@yandex.ru

Целью настоящей работы является создание способа решения задачи оптимального распределения трафика в сети с помощью контурного метода анализа данных. В первом разделе работы объяснен принцип преобразования любой доступной сети к контурному виду, причем рассмотрен случай как для сетей без потерь, так и для сетей с потерями. Во втором разделе в общем виде показан метод приведения сети в контурном виде к системе нелинейных неравенств, решив которую можно получить некое распределение трафика в системе. В заключительном разделе на примере системы массового обслуживания *М/М/1/N* показано решение задачи оптимального распределения трафика по критерию минимизации потерь. В качестве исходных данных для задачи выступили матрица инцидентности, интенсивность обслуживания и размерность буфера для каналов связи. Указанный подход по оптимальному распределению трафика позволяет сократить число используемых переменных по сравнению с известными методами на основе беспетельных марирутов, а также не требует их предварительного поиска, так как они определяются из размерности матрицы иниидентности графа моделируемой сети.

Ключевые слова: контурный метод, оптимизация трафика в сети.

Using of the contour method to solving the problem of optimal traffic distribution in the network

K. E. Gaipov, A. A. Kuznetsov, I. L. Krikunov

Reshetnev Siberian State University of Science and Technology 31, Krasnoyarskii Rabochii prospekt, Krasnoyarsk, 660037, Russian Federation E-mail: gaipovke@yandex.ru

The purpose of this work is to create a method for solving the problem of optimal traffic distribution in a network using the contour data analysis method. In the first section of the work, the principle of converting any available network to a contour form is explained, and the case is considered both for networks without loss and for networks with losses. The second section shows in a general way the method of bringing the network in contour form to a system of non-linear inequalities, by solving which one can obtain a certain distribution of traffic in the system. In the final section, using the M/M/1/N queuing system as an example, the solution of the problem of optimal traffic distribution according to the loss minimization criterion is shown. The initial data for the task were the incidence matrix, service intensity and buffer dimension for

communication channels. A feature of the proposed algorithm is the search for a contour matrix, for the compilation of which it is proposed to use loss edges as elements of the spanning tree of the graph, which allows you to immediately determine the contour matrix using the concept of a fundamental cycle of a graph. This approach to optimal traffic distribution reduces the number of variables used compared to the known methods based on loopless routes, and also does not require their preliminary search, since they are determined from the dimension of the incidence matrix of the simulated network graph.

Keywords: contour method, network traffic optimization.

Introduction

Currently, when network technologies are developing at a rapid pace, and the number of devices in the network is steadily growing, telecom operators are faced with the task of increasing the transmitted traffic quality and flexible balancing of incoming traffic without loss of network performance. Changing the bandwidth of communication channels, adding new channels and nodes to the topology leads to a complete recalculation of routing tables.

One of the ways to improve the quality of network functioning is to determine the optimal data transmission routes that implement a routing policy in which one or more optimality criteria are achieved. Such criteria can be the minimum delay and/or loss in the transmission of information both for individual routes or channels and the total, thereby achieving a more balanced load of communication channels.

Modern dynamic routing protocols, such as OSPF [1] and IS-IS [2], allow bypassing busy communication channels to improve the quality of network services. However, with frequent route updates, additional computing resources are needed to activate and install the new route. Multipath routing protocols, as a rule, apply the k-shortest paths algorithm (Yen's algorithm) used for technical solutions [3]. As an example, we can consider the MPLS-TE module [4; 5], which works with routes calculated using the above algorithm. This allows balancing the load by distributing network traffic along backup routes.

In addition, there are many upgrades and customisations of these methods. Thus, in [6-9] multipath routing models with load balancing based on GERT networks and minimizing planimetric delays for each type of traffic are proposed. In [10] an adaptive routing algorithm on neural networks is studied. In [11] a method involving load balancing with multipath routing is considered, based on the algorithm of paired route permutations, in [12; 13] the rapid IP FRR routing technologies are analyzed.

Despite an impressive list of works and various technical implementations in the form of computer programs [14-17], existing mechanisms do not solve the problem of optimal traffic in terms of losses. That is why this paper has developed and described a mathematical model of the network applying the contour method of traffic analysis by the criterion of minimizing losses.

1. Bringing the network to a contour view

Suppose we are given a network where the packet loss probability function is known for each communication channel. In the most trivial case, this functional dependence will be determined by the size of the buffer of the telecommunication device to which the communication channel is connected, the data transfer rate of the communication channel, and the statistical characteristics of the flow passing through it. If this is possible, then as such a functional dependence we can use already existing formulas describing the probability of losses for various types of queuing systems. In reality, the dependence of flow losses depends on many factors that are determined by the logic of the software and hardware of a telecommunications device.

To obtain a mathematical model of traffic distribution over the network, it is convenient to represent the network in the form of a directed graph. We define the construction rule: each direction of data transmission between a pair of telecommunication devices will be a directed edge of the graph, the models of telecommunication nodes will correspond to the vertices of the graph. One additional

edge will also exit from each node; this edge is a channel through which traffic, lost as a result of buffer overflow of telecommunication devices, will be transmitted. Fig. 1 shows a network consisting of two switching nodes SN1 and SN2. Users who simultaneously receive and transmit data are connected to SN1; in the figure they are designated as sourse and receiver SR1. Similarly, SR2 is connected to SN2. Each channel in Figure 1 is duplex, losses occur at each switching node.



Based on the above, Fig. 2 shows a network graph where nodes 1 and 2 correspond to traffic sources; edges 1 and 2 are channels connecting the corresponding sources to switching nodes. Nodes 3 and 4 correspond to the traffic receiver; edges 3 and 4 are channels connecting the corresponding receivers to the switching nodes. Thus, pairs of nodes 1 and 3 correspond to SR1; edges 1 and 3 correspond to a duplex channel between SR1-SN1; a pair of nodes 2 and 4 corresponds to SR2, edges 2 and 4 correspond to a duplex channel between SR2-SN2. Node 7 terminates losses from incoming channels to node 5, node 8 terminates losses from incoming channels to node 6, and edges numbered 7 and 8 are traffic reset channels connecting switching nodes with reset termination nodes. Nodes 5 and 6 denote the switching nodes SN1 and SN2, respectively. If channels 7 and 8 as well as nodes 7 and 8 are removed from the network in Fig. 2, then the resulting network will be a lossless network (Fig. 3).



Рис. 2. Представление сети с потерями в виде графа Fig. 2. Graph representation of a lossy network

A contour network is a network that does not contain open circuits. Any network can be represented as a contour, ensuring the circulation of the flow in the network. To ensure the flow circulation in the network, it is necessary to close all nodes that have only one edge incident, combining them into one node. This guarantees the equality of the sum of flows entering the node and the sum of flows leaving the node. Thus, for the example considered in Fig. 2 and 3 we need to combine nodes I through 4, as well as 7 and 8 (for a lossy network) into one common node. Graphically, such a network is shown in Fig. 4 and 5, respectively.



Рис. 3. Представление сети без потерь в виде графа

Fig. 3. Graph representation of a lossless network



Рис. 4. Результат приведения сети с потерями к контурному типу

Fig. 4. The result of reducing a lossy network to a contour form



Рис. 5. Результат приведения сети без потерь к контурному типу

Fig. 5. The result of reducing a lossless network to a contour form

As can be seen in Fig. 4, nodes 1 through 4 plus nodes 7 and 8 were combined into node 0, and nodes 1 through 4 were combined for the network in Fig. 5. For the resulting networks, it is necessary to determine the contour matrix.

Note that a source is not a single source of information, but a set of independent sources, for example, a PC on a local network. Thus, statistically multiplexed flows pass through the channel connected to the sources in Fig. 1.

2. Contour method of data analysis

We enter variables that will be used when composing mathematical models:

- V the set of all vertices of the graph;
- |V| the number of vertices of the graph;
- E the set of all edges of the graph;
- |E| the number of edges of the graph;

 E_h – the subset of the edges of the graph that are chords;

 $|E_h|$ – the number of edges in the set E_h ;

 E_b – the subset of the edges of the graph that are branches;

 $|E_h|$ – the number of edges in the set E_h ;

 E_{src} – the subset of the edges of the graph to which information sources are connected;

 $|E_{src}|$ – the number of edges in the set E_{src} ;

 E_{rcv} – the subset of the edges of the graph to which the recipients of information are connected;

 $|E_{rcv}|$ – the number of edges in the set E_{rcv} ;

 E_{tr} – the subset of the edges of the graph that connect nodes that are switching nodes;

 $|E_{tr}|$ – the number of edges in the set E_{tr} ;

 E_{loss} – te subset of the edges of the graph, which are the channels of information reset;

 $|E_{loss}|$ – the number of edges in the set E_{loss} .

According to graph theory, any graph can be described by an incidence matrix; the incidence matrix without a linearly dependent row is a matrix of linearly independent sections, which in turn is orthogonal to the contour matrix [18].

$$\mathbf{I}_{\left(|V|-1\right)\times|E|} \cdot \mathbf{C}_{|E_b|\times|E|}^{T} = 0, \qquad (1)$$

where $\mathbf{I}_{(|V|-1) \times |E|}$ is the incidence matrix of a contour network without a linearly independent row;

 $C_{|E_b| \times |E|}$ is the contour matrix.

The matrix $\mathbf{I}_{(|V|-1)\times|E|}$ is formed from the incidence matrix $\mathbf{I}_{|V|\times|E|}$, describing the graph of the original orthogonal network by removing from it all rows in which there is only one non-zero element. Thus, the transformation of an orthogonal network into a contour network reduces so that all nodes with a degree equal 1 and incident edges from subsets E_{src} , E_{rcv} and E_{loss} are combined into one node. This is equivalent to adding all the rows in the matrix $\mathbf{I}_{|V|\times|E|}$, in which there is only one non-zero element in one row. Thus, since the new matrix is the incidence matrix of the contour network, where

one of the rows can also be distinguished as linearly dependent, then the row that resulted from the addition of rows with one non-zero element, can be chosen thereof. The resulting incidence matrix, by adding rows and rearranging columns, is reduced to the following form:

$$\mathbf{I}_{(|V|-1)\times|E|} = \left[\mathbf{E} \ \mathbf{Hg}_{(|V|-1)\times|E_h|}\right].$$

where **E** is the unity matrix of dimension |V|-1, $\mathbf{Hg}_{(|V|-1)\times|E_h|}$ is the matrix of the chords of the graph.

Note that the column numbers forming the unity matrix are the numbers of the graph branches that represent the network analyzed, while the column numbers of the matrix **Hg** are the numbers of the chords of this graph.

To fulfill the orthogonality condition according to formula (1), the matrix $C_{|E_b| \times |E|}$ must have the following form:

$$\mathbf{C}_{|E_b|\times|E|} = \left[\mathbf{E} \left(-\mathbf{H}\mathbf{g}_{(|V|-1)\times|E_h|}^{\mathsf{T}}\right)\right].$$

A feature of the lossy networks topology is that the spanning tree of a graph can be obtained only from edges that form reset channels: these edges will be branches of the graph, while the remaining edges will be chords of this graph.

Thus, it is possible to propose a faster way to obtain a contour matrix, which requires less computational steps and is applicable when necessary to analyze losses in all channels or in most communication channels.

We introduce the following notations:

 \mathbf{I}_{loss} – incidence matrix of the lossy network, reduced to a contour form;

 $I_{lossless}$ – lossless incidence matrix reduced to a contour view;

Xh – column vector, with each element showing flows in chords from each source;

 N_{users} – number of sources equal to the number of elements in the set E_{src} ;

n – source number.

In the matrix \mathbf{I}_{loss} and $\mathbf{I}_{lossless}$ there are the same number of nodes, but a different number of columns, therefore, the matrix \mathbf{I}_{loss} is obtained from the matrix $\mathbf{I}_{lossless}$ by adding $|E_{loss}|$ columns to the left, while the columns being added are a diagonal unity matrix, which corresponds to the following expression:

$$\mathbf{I}_{loss} = \begin{bmatrix} \mathbf{E} \ \mathbf{I}_{lossless} \end{bmatrix}$$

This implies that the matrix $\mathbf{I}_{lossless}$ is a chord matrix for a graph describing a lossy network. Thus, in the matrix \mathbf{I}_{loss} columns numbered from 1 to $|E_{loss}|$ will be branches of the graph, and columns from $(|E_{loss}|+1)$ to $(|E_{loss}|+|E_h|)$ will be chords of this graph. Therefore, the contour matrix for a lossy network will be defined as follows:

$$\mathbf{C}_{loss} = \left[\mathbf{E} \left(-\mathbf{I}_{lossless}^{T} \right) \right].$$

In the matrix C_{loss} columns numbered from 1 to $|E_h|$ will be the chords of the graph, columns from $(|E_h|+1)$ to $(|E_h|+|E_{loss}|)$ are branches of this graph.

We formulate recommendations on the choice of edge numbers. Based on the notation entered, the recommended numbering order of the edges in the original graph is as follows: first, to number all the edges from the set E_{src} , then from E_{rcv} , followed by E_{tr} and finally from E_{loss} . This recommendation is due to the fact that in the future there will be no need to organize additional search and sorting algorithms necessary for the formation of the final system of constraints and the objective function.

Each row in the contour matrix C_{loss} shows which edges are included in the contour, whereas only one chord is included in each contour, and the direction of this chord sets the direction of the entire contour. Therefore, if the values of the flows in each chord of the graph are known and these values are set as a column vector, then the product of the transposed contour matrix by the vector of flows in the chord will give the values of flows in both each chord and the branches of the graph, thereby the flows in the branches of the graph are linearly dependent on the flows in the chords of this graph or, similarly, flows in the edges of a graph are completely described by flows in its chords. Thus, in the compiled mathematical model, the flows in the chords of the graph will act as variables.

Since each source creates its own flow in each edge of the graph, the number of variables will be determined as the number of chords of the graph multiplied by the number of traffic sources.

Consequently, the vector of variables Xh will contain $N_{users} \cdot |E_h|$ variables. In the vector Xh variables with ordinal numbers from $(n-1)\cdot(|E_h|+1)$ to $(n\cdot|E_h|)$ will show the source streams n in chords from the set E_h . To determine the total flow in each communication channel, we will create the following matrix:

$$\mathbf{C}_{sum} = \begin{bmatrix} \mathbf{C}_{loss1}^T & \mathbf{C}_{loss2}^T & \cdots & \mathbf{C}_{lossN_{users}}^T \end{bmatrix}.$$

To determine the flow in each channel created by each source, we will make the following matrix:

$$\mathbf{C}_{users} = \begin{bmatrix} \mathbf{C}_{loss1}^{T} & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{C}_{lossN_{users}}^{T} \end{bmatrix}.$$
 (2)

To determine the total flow in each communication channel

$$Xe_{sum} = \mathbf{C}_{sum} \cdot Xh \,. \tag{3}$$

Each row of the matrix Xe_{sum} shows the total flow in each communication channel. To determine the flow in each channel from each source, use the formula

$$Xe_{users} = \mathbf{C}_{users} \cdot Xh$$

Each row *m* of the matrix Xe_{users} shows flows from each individual user. Thus, to find out which flow is being created in the edge E_i , it is necessary to take an element $Xe_{users(n-1)|E|+i}$ from the user *n*. Having obtained the connections of the flows in each channel through the flows in the chords, the further task is reduced to compiling a system of constraints and an objective function.

Since the dimension of such a problem is large enough, it is necessary to use mathematical modeling systems to solve it. To form a system of constraints, it is necessary to make a number of matrices **A**, **B**, **Aeq**, **Beq**, **Ceq**(x) and **C**(x), and also set the point of the initial iteration x0. A system of nonlinear inequalities is created from the obtained matrices

$$\begin{cases} \mathbf{A} \cdot X \leq 0; \\ \mathbf{Aeq} \cdot X = 0; \\ \mathbf{C}(x) \leq 0; \\ \mathbf{Ceq}(x) = 0. \end{cases}$$

As can be seen, the system consists of four blocks, where Ceq(x) is a column vector, each element of which represents a sort of nonlinear dependence on the desired variables, while this value should be zero. Vector C(x) is similar to block Ceq(x), however, by contrast each element of this column vector is less than or equal to zero.

To determine the value of the flow in the lossy channels, we will leave in the matrix C_{loss} only those columns that are responsible for the edges from the set E_{loss} (i.e. columns with numbers corresponding to the edges through which the lossy flows pass). We denote such a matrix as $C(E_{loss})_{users}$ and form as follows

$$\mathbf{C}(E_{loss})_{users} = \begin{bmatrix} \mathbf{C}(E_{loss})_{loss1}^{T} & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{C}(E_{loss})_{lossN_{users}}^{T} \end{bmatrix}$$

To determine the lossy flow in each lossy channel from each source, we write the following equation:

$$Xe(E_{loss})_{users} = \mathbf{C}(E_{loss})_{users} \cdot Xh.$$
(4)

By formula (4) we obtain the values of lossy flows expressed in terms of flows in the chords of the graph. On the other hand, the loss in the source loss channel can be defined as the traffic intensity in the channel that enters the switching node multiplied by the probability of reset for this channel.

Further we need to understand for which channels it is necessary to determine the losses: in case of all channels, the space of channels from subsets belongs to channels E_{src} , E_{rcv} and E_{tr} ; in case we are interested in channels connecting switching nodes only, then this is the space of channels from the subset E_{tr} . The latter option is of the greatest practical interest since the edges from the subset E_{tr} , as a rule, are the edges of the backbone part of the network, while the edges from the subsets E_{src} and E_{rcv} belong to the access channels and are poorly loaded.

To determine the probability of losses in each channel from the subset E_{tr} it is necessary to determine the total flow in this set. To do this, we will leave only those columns in the matrix C_{loss} that correspond to the numbers of edges from the subset E_{tr} and denote such a matrix as $C(E_{tr})_{loss}$. To determine the total flow in the edges of the subset E_{tr} we make a matrix $C(E_{tr})_{sum}$

$$\mathbf{C}(E_{tr})_{sum} = \begin{bmatrix} \mathbf{C}(E_{tr})_{loss1}^{T} & \mathbf{C}(E_{tr})_{loss2}^{T} & \cdots & \mathbf{C}(E_{tr})_{lossN_{users}}^{T} \end{bmatrix}.$$

By analogy with (2), the flow in each channel from the subset E_{tr} created by each source is expressed as follows:

$$\mathbf{C}(E_{tr})_{users} = \begin{bmatrix} \mathbf{C}(E_{tr})_{loss1}^{T} & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{C}(E_{tr})_{lossN_{users}}^{T} \end{bmatrix}.$$

Next, we find the value of the flow vectors for a subset of edges E_{tr} :

$$Xe(E_{tr})_{sum} = \mathbf{C}(E_{tr})_{sum} \cdot Xh, \qquad (5)$$

$$Xe(E_{tr})_{users} = \mathbf{C}(E_{tr})_{users} \cdot Xh, \qquad (6)$$

where $Xe(E_{tr})_{sum}$ shows the total flow passing through the edges of the subset E_{tr} , the element of this vector shows the value of the total flow in the *i*th edge; $Xe(E_{tr})_{users}$ shows the flows from each source passing through the edges of the subset E_{tr} .

Each row $|E_{tr}|$ of the matrix $Xe(E_{tr})_{users}$ shows the flows from each source in the edges of the subset E_{tr} . Thus, to find out which flow is being created in the *i*th edge from the user *n*, it is necessary to take an element $Xe(E_{tr})_{users(n-1)|E_{tr}|+i}$. If the probability function of losses for a channel from a subset E_{tr} is known, the intensity of losses in the channel for the source *n* will be as follows

$$Xe(E_{tr})_{users(n-1)|E_{tr}|+i} \cdot p(Xe(E_{tr})_{sumi}).$$

The flow in the reset channel from the source n, which is connected to the node v_i , is equal to the sum of the flows along the edges that enter it, corresponding to such edges for which -1 is in the incident matrix in the row indexing the v_i node. If for each node to which the reset channels are connected the flows from the incoming edges are summed, then such a flow will correspond to the reset flow from a specific source in the reset channel originating from this node:

$$Xe_{nv} = \sum_{i \in v(in)} Xe(E_{tr})_{users(n-1)|E_{tr}|+i} \cdot p(Xe(E_{tr})_{sumi}),$$
(7)

where Xe_{nv} is the discharge flow at the node v_i from the source *n*.

If all the elements Xe_{nv} are put in the order of increasing the index, the value of the elements of the resulting vector Xe_{nv} will be equal to the elements of the vector $Xe(E_{loss})_{users}$, thus $Xe_{nv} = Xe(E_{tr})_{users}$, with the implication that

$$\mathbf{Ceq}(x) = Xe(E_{tr})_{users} - Xe_{nv}.$$

Matrix C(x) can be used to add third-party restrictions, for example, add restrictions on losses in communication channels, set delays for channels or end-to-end delays, and also leave the matrix empty for cases when additional conditions are not required.

For loss restrictions, it is necessary to create a column vector P_{losses} , in which we specify the threshold value of loss probabilities for individual channels or all channels in the network. Knowing the functional dependence of the probability of losses for these channels, as well as using the total intensity of losses in these channels from formula (3), we obtain the following expression:

$$p(Xe_{sum}) \le P_{losses} \,. \tag{8}$$

By the same principle, we will create a column vector T_{wt} , in which we will specify the delay thresholds for channels or through delays for certain routes. Knowing the functional dependence of the average number of applications in the queue L_0 , as well as using the total intensity of losses in these channels from formula (3), we obtain an expression for channel and end-to-end delays:

$$L_0(Xe_{sum}) / Xe_{sum}$$
.

By analogy with the expression (8), we make up inequalities for a system of constraints:

$$L_0\left(Xe_{sum}\right) / Xe_{sum} \le T_{wt} \,. \tag{9}$$

When forming C(x), it should be taken into account that each element of this set must be less than or equal to zero. To achieve this condition, in formulas (8) and (9), the right part of the inequality is transferred to the left. Then the final form of the column vector C(x) is the following:

$$C(x) = \begin{bmatrix} p(Xe_{sum}) - P_{losses} \\ L_0(Xe_{sum}) / Xe_{sum} - T_{wt} \end{bmatrix}$$

Further we consider the algorithm for obtaining a matrix Aeq: the product of a matrix Aeq by a vector of variables X describes such a linear combination of values that are known in advance, which according to the mathematical model are flows in the branches of a subset E_{rcv} . It should also be taken into account that the flows from the source n will be zero in all edges of the space E_{src} except for the edge directly connected to the source n.

Therefore, if only the rows corresponding to the edge numbers from the subset E_{rcv} are left from the matrix \mathbf{C}_{loss} , as well as the rows corresponding to the edge numbers of the subset E_{src} , in which the flows from the sources are zero, and also denoting such a space as E_{S0} , and denoting the resulting matrix $\mathbf{C}(E_{rcv} \cup E_{S0})_{loss}$, we obtain the following matrix:

$$\mathbf{C}(E_{rcv} \cup E_{S0})_{users} = \begin{bmatrix} \mathbf{C}(E_{rcv} \cup E_{S0})_{loss1}^{T} & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{C}(E_{rcv} \cup E_{S0})_{lossN_{users}}^{T} \end{bmatrix}.$$

If we multiply the resulting matrix with the vector Xh, we obtain the vector $Xe(E_{rcv} \cup E_{S0})_{users}$, describing the flows in channels that are quantities known in advance, and their values are entered in the vector **Beq**. Thus,

$$\mathbf{Aeq} = \mathbf{C} \left(E_{rcv} \cup E_{S0} \right)_{users},$$
$$Xe \left(E_{rcv} \cup E_{S0} \right)_{users} = \mathbf{C} \left(E_{rcv} \cup E_{S0} \right)_{users} \cdot Xh.$$

We consider the algorithm of A matrix formation. The matrix A is formed due to the necessary to set limits for the flows that pass through the communication channels. The minimum desired restriction is that the streams in each channel created by each source should not be negative, since the value of the streams in the edges of the subset E_{rcv} are known in advance, and the value of the flows in the edges of the subset E_{loss} depends on the values of the flows in the edges of the subset E_{src} and E_{tr} . In the matrix, it is necessary to leave only those columns with the numbers corresponding to the numbers of edges from the space $(E_{src} - E_{S0})$ and E_{tr} . The result is a matrix $C(E_{src} - E_{S0} \cup E_{tr})_{loss}$ from which the matrix $C(E_{src} - E_{S0} \cup E_{tr})_{users}$ is formed:

$$\mathbf{C}(E_{src} - E_{S0} \cup E_{tr})_{users} = \begin{bmatrix} \mathbf{C}(E_{src} - E_{S0} \cup E_{tr})_{loss1}^{T} & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{C}(E_{src} - E_{S0} \cup E_{tr})_{lossN_{users}}^{T} \end{bmatrix}.$$

This patrics matrix is the matrix \mathbf{A} , the value of the elements of the vector \mathbf{B} equals 0. Thus, a system of constraints is obtained. Since the number of equalities in this system is less than the number of variables, this system has an infinite set of solutions, and each solution gives one of the possible traffic distributions along the chords of the graph, after which, by multiplying the resulting vector by the \mathbf{C}_{loss} matrix, we obtain the final values of flows in all edges of the graph.

To find the best solution, an optimality criterion should be set, which can be expressed using various functions. If we use network losses as a criterion, i.e. minimize the sum of the reset streams, the objective function will be the sum of the array elements $Xe(E_{loss})_{users}$ or Xe_{nv} and will be as follows:

$$f(x) = \sum_{i=1}^{|E_{loss}|} Xe(E_{loss})_i.$$

3. An example of bringing a network to a system of restrictions

As an example, we consider the network shown in Fig. 6. Denote the value of the variables for this topology, which we will use: $V = \{1, 2, 3, 4\}$, |V| = 4, $E = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13\}$, |E| = 13, $E_h = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$, $|E_h| = 10$, $E_b = \{11, 12, 13\}$, $|E_b| = 3$, $E_{src} = \{3, 4\}$, $|E_{src}| = 2$, $E_{rcv} = \{1, 2\}$, $|E_{rcv}| = 2$, $E_{tr} = \{5, 6, 7, 8, 9, 10\}$, $|E_{tr}| = 6$, $E_{loss} = \{11, 12, 13\}$, $|E_{loss}| = 3$.



Рис. 6. Исследуемая сеть Fig. 6. Researched network

At the next stage, we will define the contour matrix C_{loss} (Table 1).

According to the given topology, the number of sources is 2, hence the variable $N_{users} = 2$, and the vector of variables Xh will contain $|E_h| \cdot N_{users} = 10 \cdot 2 = 20$ values. The elements of the vector Xh contain unknown variables, for example, variables x_1 and x_{11} show the speed value of information flows in edge No. 1 created by two different sources. Next, according to the algorithm, we determine the values Xe_{sum} by formula (5) and Xe_{users} by formula (6). Vector Xe_{sum} shows which components make up the flow in the corresponding edges. The vector elements Xe_{users} show the value of the flow created by a single source in the corresponding edge.

Table 1

E _{rcv}		E _{src}		E _{tr}					Eloss			
1	2	3	4	5	6	7	8	9	10	11	12	13
1	0	0	0	0	0	0	0	0	0	-1	0	0
0	1	0	0	0	0	0	0	0	0	0	-1	0
0	0	1	0	0	0	0	0	0	0	1	0	0
0	0	0	1	0	0	0	0	0	0	0	1	0
0	0	0	0	1	0	0	0	0	0	-1	1	0
0	0	0	0	0	1	0	0	0	0	1	-1	0
0	0	0	0	0	0	1	0	0	0	-1	0	1
0	0	0	0	0	0	0	1	0	0	1	0	-1
0	0	0	0	0	0	0	0	1	0	0	1	-1
0	0	0	0	0	0	0	0	0	1	0	-1	1

Contour matrix for the network under study

Further it is necessary to determine the nonlinear part of the equations of the mathematical model: we define a matrix $\mathbf{C}(E_{loss})_{loss}^{T}$ this matrix consists of rows defined by rows of a subset E_{loss} from the matrix \mathbf{C}_{loss}^{T} . Then we determine the value of the lossy flows in the loss channels from each source by the formula (4). On the other hand, the value of lossy flows depends on which flow passes through the edges of the set E_{tr} . Thus, it is possible to determine the proportion of the information flow lost for each edge as the product of the flow in the edge from a specific source by the probability of losses for this edge. To do this, we determine the value of the flow vectors for the edge space E_{tr} . In order to use formula (5), we find $C(E_{tr})_{sum}$ and $C(E_{tr})_{users}$ to use in formula (6). Employing matrics $C(E_{tr})_{sum}$ and formula (5), we find the value of the total flows in the edges of the set E_{tr} , while using the matrix $C(E_{tr})_{users}$ and formula (6), we find the value of the flows in the edges from the set E_{tr} for each individual source.

Further, using the formula (7), we determine the value of lossy flows for each node in the transport network, which includes flows going through a subset of the E_{tr} . We believe that the functional dependence of the probability of losses is known, and the final probability for the total flow is equal to the sum of the probabilities of losses for any component of such a flow. As a result, the value of the vector Ceq(x) or the first system of equations of the mathematical model of traffic distribution will look as follows:

$$\begin{pmatrix} (-x_1 + x_3 - x_5 + x_6 - x_7 + x_8) - (x_6 \cdot P(x_6 + x_{16}) + x_8 \cdot P(x_8 + x_{18})) = 0 \\ (-x_2 + x_4 + x_5 - x_6 + x_9 - x_{10}) - (x_5 \cdot P(x_5 + x_{15}) + x_9 \cdot P(x_9 + x_{19})) = 0 \\ (+x_7 - x_8 - x_9 + x_{10}) - (x_7 \cdot P(x_7 + x_{17}) + x_{10} \cdot P(x_{10} + x_{20})) = 0 \\ \end{pmatrix}$$

$$\begin{pmatrix} (-x_{11} + x_{13} - x_{15} + x_{16} - x_{17} + x_{18}) - (x_{16} \cdot P(x_6 + x_{16}) + x_{18} \cdot P(x_8 + x_{18})) = 0 \\ (-x_{12} + x_{14} + x_{15} - x_{16} + x_{19} - x_{20}) - (x_{15} \cdot P(x_5 + x_{15}) + x_{19} \cdot P(x_9 + x_{19})) = 0 \\ (+x_{17} - x_{18} - x_{19} + x_{20}) - (x_{17} \cdot P(x_7 + x_{17}) + x_{20} \cdot P(x_{10} + x_{20})) = 0 \\ \end{pmatrix}$$

The next step is to find a mathematical model of linear equalities. To do this, we define the matrix **Aeq**, for which we define the matrices $C(E_{rcv} \cup E_{S0})_{loss1}$ and $C(E_{rcv} \cup E_{S0})_{loss2}$, respectively:

Since it is known that $Aeq = C(E_{rcv} \cup E_{S0})_{users}$, then by formula (8) it is possible to make a matrix **Aeq**. To obtain the system of equations itself, we determine the value of the product **Aeq** by a vector *Xh*, denote the resulting vector by *Xeq*, and also introduce a column vector **Beq**, that shows the value of each element of the vector *Xeq*. As a result, we obtain the following equality:

$\operatorname{Aeq} \cdot Xeq = \operatorname{Beq} .$

We establish that the flow created by the source connected to edge No. 3 should deliver 10 pieces of information to the recipient connected to edge No. 2. Obviously, the source does not send anything to itself, so the value of the flow from this source in edge No.1 will equal 0, and also this source cannot create a stream in edgeNo. 4, so the flow in this edge is also equal to 0, whereas the value of the flow itself in the source is unknown, since the loss flows in No. 11, 12 and 13 edges are unknown. From the same considerations, the distribution of the flow from the second source is constructed: the value of the flow in the edge No. 1 is equal to 13 units, in the edges No. 2 and 3 are 0, and the value in the edge No. 4 is unknown.

The final basic constraint is that the intensity of data flows must be non-negative, hence for this we need to find a matrix $\mathbf{A} = \mathbf{C} (E_{src} - E_{S0} \cup E_{tr})_{loss}$ and multiply it by a vector Xh, resulting in a vector

 $Xh(E_{src} - E_{S0} \cup E_{tr})_{loss}$. However, in this example, we can use a unit matrix **E** of dimension $|E_h| \cdot N_{users} = 20$ instead of a matrix **A**.

To obtain the system of equations itself, we determine the value of the product **A** of the vector Xh, denote the resulting vector for Xe, and also introduce a column vector **B**, that shows the value of each element of the vector Xe. As a result, we obtain the following equality: $\mathbf{A} \cdot Xe \ge \mathbf{B}$. The matrix **B** in this example is given by a column vector entirely consisting of zeros. The number of elements in the matrix **B** is equal to the number of chords multiplied by the number of sources N_{users} , i.e $10 \cdot 2 = 20$.

Applying this method, we fulfill the condition of non-negativity of information flows in the network, although we add additional linear inequalities to the system. It is important to note that this method should not be used for large-dimensional topologies, where unnecessary inequalities can affect the performance of the solution.

The last stage in the formation of a mathematical model is to obtain an objective function, which is defined as the sum of the lossy flows, i.e. it is the sum of the rows of the matrix $Xe(E_{loss})_{users}$. As a result, the objective function is the difference between the flows that entered the network through the channels of space E_{src} and the flows that left the network along the edges of space E_{rcv} :

$$F = x_3 + x_{13} + x_4 + x_{14} - (x_1 + x_{11} + x_2 + x_{12}).$$

We bring the obtained constraints and the target function into a single system:

$$\begin{cases} \left(-x_{1} + x_{3} - x_{5} + x_{6} - x_{7} + x_{8}\right) - \left(x_{6} \cdot P(x_{6} + x_{16}) + x_{8} \cdot P(x_{8} + x_{18})\right) = 0; \\ \left(-x_{2} + x_{4} + x_{5} - x_{6} + x_{9} - x_{10}\right) - \left(x_{5} \cdot P(x_{5} + x_{15}) + x_{9} \cdot P(x_{9} + x_{19})\right) = 0; \\ \left(+x_{7} - x_{8} - x_{9} + x_{10}\right) - \left(x_{7} \cdot P(x_{7} + x_{17}) + x_{10} \cdot P(x_{10} + x_{20})\right) = 0; \\ \left(-x_{11} + x_{13} - x_{15} + x_{16} - x_{17} + x_{18}\right) - \left(x_{16} \cdot P(x_{6} + x_{16}) + x_{18} \cdot P(x_{8} + x_{18})\right) = 0; \\ \left(-x_{12} + x_{14} + x_{15} - x_{16} + x_{19} - x_{20}\right) - \left(x_{15} \cdot P(x_{5} + x_{15}) + x_{19} \cdot P(x_{9} + x_{19})\right) = 0; \\ \left(-x_{12} - x_{18} - x_{19} + x_{20}\right) - \left(x_{17} \cdot P(x_{7} + x_{17}) + x_{20} \cdot P(x_{10} + x_{20})\right) = 0; \\ \left(+x_{17} - x_{18} - x_{19} + x_{20}\right) - \left(x_{17} \cdot P(x_{7} + x_{17}) + x_{20} \cdot P(x_{10} + x_{20})\right) = 0; \\ Xh_{i} \ge 0(i = 1, 2, \dots, 20); \\ x_{1} = 0; \\ x_{2} = 10; \\ x_{2} = 10; \\ x_{11} = 15; \\ x_{12} = 0; \\ x_{13} = 0; \\ F = x_{3} + x_{13} + x_{4} + x_{14} - \left(x_{1} + x_{11} + x_{2} + x_{12}\right) \rightarrow min. \end{cases}$$

To find a numerical solution, it is necessary to set the value of the loss probability function: in the classical version, it can be assumed that the streams created by the sources have an exponential distribution of intervals between calls; the service time is also distributed exponentially and it is known that there will also be an exponential flow at the output during maintenance [19]. It is also possible to make an assumption about independence of package maintenance by each queuing system. Thus, to describe the probability we can use the formula for a queuing system M/M/1/N at a low loss level. The discharge intensity formula for the M/M/1/N system has the following form:

$$p(\lambda_{sum}) = \frac{1 - \frac{\lambda_{sum}}{\mu}}{1 - \left(\frac{\lambda_{sum}}{\mu}\right)^{N+1}} \cdot \left(\frac{\lambda_{sum}}{\mu}\right)^{N},$$

where λ_{sum} is the total intensity of traffic flow through the single-server queueing system;; μ is the intensity of traffic service; (N-1) is the number of places in the buffer.

The values of service intensities in the edges of the space E_{tr} for the analyzed network equal $\mu_5 = 15, \mu_6 = 15, \mu_7 = 20, \mu_8 = 20, \mu_9 = 25, \mu_{10} = 25$, the values of waiting places for the space are $E_{tr} - N_5 = 5, N_6 = 4, N_7 = 4, N_8 = 1, N_9 = 3, N_{10} = 8$.

To solve this optimization problem numerically, it is necessary to involve third-party tools that allow solving systems of nonlinear equations using one or more optimality criteria. For example, you can use the MatLab environment with the Optimization Toolbox add-on package connected. It is also possible to use free analogues: GNU Octave with the Optim package connected, or the Python programming language with the numpy and scipy libraries. For this example, use the MatLab environment and the linprog and fmincon functions. The full programming code is written in [20].

When solving optimization problems, it is necessary to find the point of the initial iteration by solving a single system of constraints without an optimality criterion using the linprog function.

Solving this optimization problem with the fmincon function, we obtain the following distribution, shown in Table. 2. The total traffic distribution is shown in Fig. 7.

Table 2

Edge number	The value of flows from the source No. 1 (inf.unit/s)	The value of flows from the source No. 2 (inf.unit/s)	λ _{sum} (inf.unit/s)	µ _{обслуж} (inf.unit/s)	Number of waiting places	Edge space	
1	0	13	13	_	_	Flows in receiver	
2	10	0	10	_	_	channels, E_{rcv}	
3	10.055	0	10.055	—	_	Flows in source	
4	0	14.609	14.609	-	-	channels, E_{src}	
5	5.6507	0	5.6507	15	5		
6	0	10.448	10.448	15	4		
7	4.4038	0	4.4038	20	4	Flows in transit	
8	0	4.1618	4.1618	20	1	channels, E_{tr}	
9	4.3958	0	4.3958	25	3		
10	0	4.1618	4.1618	25	8		
11	0	1.6094	1.6094	_	_	Flows in the discharge	
12	4.6511×10^{-2}	0	4.6511×10^{-2}	_	_	channels <i>E</i> .	
13	8.077×10^{-3}	2.046×10^{-6}	8.0791×10^{-3}	_	_	chamiers, <i>L</i> loss	

Distribution of flows across communication channels



Рис. 7. Суммарное распределение потоков в исследуемой сети

Fig. 7. The total distribution of flows in the studied network

In Fig. 7, in a rectangle with a background fill, the first digit shows the number of requests coming to the interface, the second digit shows the bandwidth of the interface (service intensity), and the third digit shows the size of the interface queue. Thus, for channel 7 we have 4.4038/20/4. This means that the traffic intensity on the channel makes 4.4038 packets/s, the service intensity equals 20 packets/s, and the buffer size is 4.

A numerical example shows that for given throughput capacities and buffer sizes, the network serves a given intensity of requirements entering the network with small overloads, as evidenced by insignificant losses in the reset channels.

It should be noted that the solution of the constraint system in general cannot be used independently to solve the problem of traffic distribution without an objective function, since such a system has an infinite number of solutions that contain loop routes. Therefore, in order to solve the problem by the contour method, in addition to the system of constraints it is necessary to use the objective function, as only in this case it is possible to obtain non–linear routes, which is the main difference from the method proposed in [21]. In [21] the problem of optimal traffic distribution in lossless networks was solved by the minimum delay criterion. As part of the task to be solved, it was necessary to determine all the non-parallel routes between each source-recipient pair, and the values of the flows along these routes are the desired variables. In [19], analysis of the growth in the number of pointless routes is justified only in small topologies or those with a small number of alternative paths between sources and recipients.

Conclusion

For the contour method with losses, the number of variables is determined by the product of the sources number and the number of chords in the graph, i.e. adding another edge for the contour method will either increase the number of linearly independent variables in the mathematical model by the number of load sources, or do not increase at all, depending on whether this edge is a chord or a branch, respectively. For optimization problems in which the compilation of a mathematical model is based on the search for pointless routes, adding another edge related to the chord will mean that the number of new variables will increase many times depending on where such an edge is added compared to the contour approach. A feature of the proposed algorithm is the search for a contour matrix, for which it is proposed to use lossy edges as elements of the spanning tree of the graph, which allows immediately

determine the contour matrix applying the concept of fundamental cycle of the graph. Thus, this approach for optimal traffic distribution allows to reduce the number of variables used in comparison with the known methods based on loopless routes, and also does not require their preliminary search since they are determined from the dimension of the incidence matrix of the simulated network graph.

Благодарности. Исследование выполнено при финансовой поддержке программы стратегического академического лидерства «Приоритет-2030» СибГУ им. М. Ф. Решетнева.

Acknowledgments. The reported study was funded by strategic academic leadership program «Priority-2030» of Reshetnev University.

Библиографические ссылки

1. Moy J. OSPF Version 2, STD 54, RFC 2328, April 1998.

2. Sridharan A., Guerin R., Diot C. Achieving near-optimal traffic engineering solutions for current OSPF/IS-IS networks // IEEE /ACM Trans. Netw. 2005. Vol. 13. P. 234–247.

3. Yen J. Y. Finding the K Shortest Loopless Paths in a Network // Management Science. 1971. Vol. 17, No. 11. P. 712–716.

4. RFC3272, Overview and Principles of Internet Traffic Engineering / D. Awduche, A. Chiu, A. Elwalid et al. [Электронный ресурс]. URL: http://www.ietf.orglrfc/rfc3272.txt.

5. MPLS-TE and MPLS VPN Switch OpenFlow / A. R. Sharafat, S. Das, G. M. Parulkar, N. McKeown // ACM SIGCOMM 2011 Conference on Applications, Technologies, Architectures and Protocols for Computer Communications, Toronto, ON, Canada. August 2011.

6. Shibanov A. P., Koryachko V. P., Izhvanov Y. L. Modeling of Aggregated Telecommunication Link with Technology of OpenFlows // Radioengineering. 2012. No. 3. P. 109–112.

7. Lemeshko A. V., Vavenko T. Y. Improvement of Flow Model of Multipath Routing on the Basis Load Balancing // Problems of telecommunications. 2012. Vol. 6, No. 1. P. 12–29.

8. Lemeshko A. V., Vavenko T. V. Development and Research of the Flow Model of Adaptive Routing in the Software-Defined Networks with Load Balancing // Proc. of Tomsk State University of Control Systems and Radioelectronics. 2013. Vol. 29, No. 3. P. 100–108.

9. Merindol P. Improving Load Balancing with Multipath Routing // Proc. of the 17-th International Conference on Computer Communications and Networks (IEEE ICCCN 2008). 2008. P. 54–61.

10. Mikhailenko V. S., Solodovnik M. S. Analysis of the Adaptive Neural Network Router // Automatic Control and Computer Science. 2016. Vol. 50, No. 1. P. 46–53.

11. Перепелкин Д. А., Цыганов И. Ю. Алгоритм парных переходов в компьютерных сетях на основе метода маршрутизации по подсетям // Вестник РГРТУ. 2016. № 57. С. 56–62.

12. Goulamghoss M. I., Bassoo V. Analysis of traffic engineering and fast reroute on multiprotocol label switching // J Ambient Intell Human Comput. 2021. No. 12. P. 2409–2420. https://doi.org/10.1007/s12652-020-02365-5.

13. Enhanced Multicast Repair Fast Reroute Mechanism for Smart Sensors IoT and Network Infrastructure/ J. Papan, P. Segec, O. Yeremenko et al. // Sensors. 2020. No. 20. P. 3428. https://doi.org/10.3390/s20123428.

14. Пат. № 2017615438 Российская Федерация. Модуль динамической балансировки потоков данных в программно-конфигурируемых сетях с обеспечением качества сетевых сервисов / Д. А. Перепелкин, В. С. Бышов. – № 2017612610 ; заявл. 22.03.2017 ; опубл. 16.05.2017.

15. Пат. № 2017612645 Российская Федерация. Модуль многопутевой маршрутизации в программно-конфигурируемых сетях на базе протокола OpenFlow / Д. А. Перепелкин, М. А. Иванчикова. – № 2016660593 ; заявл. 13.10.2016 ; опубл. 02.03.2017.

16. Пат. № 2015662938 Российская Федерация. Программа динамической межузловой балансировки трафика / Д. С. Исхоков, Э. Р. Зарипова. – № 2015618100 ; заявл. 04.09.2015 ; опубл. 20.01.2016.

17. Пат. №2015662569 Российская Федерация. Система многопутевой адаптивной маршрутизации и балансировки нагрузки в динамических корпоративных сетях / Д. А. Перепелкин, Б. С. Бышов. – № 2015619243 ; заявл. 05.10.2015 ; опубл. 20.12.2015.

18. Омельченко А. В. Теория графов. М. : МЦНМО, 2018. 416 с.

19. Алиев Т. И. Основы моделирования дискретных систем. СПб. : СПбГУ ИТМО, 2009. 363 с.

20. Пат. № 2022684654 Российская Федерация. Оптимальное распределение трафика сети массового обслуживания на основе контурного метода по критерию минимума потерь / К. Э. Гаипов, И. Л. Крикунов, А. А. Демичева. – № 2022683987 ; заявл. 08.12.2022 ; опубл. 15.12.2022.

21. Бертсекас Д., Галлагер Р. Сети передачи данных : пер. с англ. Н. Б. Лиханова и др. / под ред. Б. С. Цыбакова. М. : Мир, 1989. 544 с.

References

1. Moy J. OSPF Version 2, STD 54, RFC 2328, April 1998.

2. Sridharan A., Guerin R., Diot C. Achieving near-optimal traffic engineering solutions for current OSPF/IS-IS networks. *IEEE/ACM Trans. Netw.* 2005, Vol. 13, P. 234–247.

3. Yen J. Y. Finding the K Shortest Loopless Paths in a Network. *Management Science*. 1971, Vol. 17, No. 11, P. 712–716.

4. Awduche D., Chiu A., Elwalid A., Widjaja I., Xiao X. RFC3272, Overview and Principles of Internet Traffic Engineering. Available at: http://www.ietf.orglrfc/rfc3272.txt.

5. Sharafat A. R., Das S., Parulkar G. M., McKeown N. MPLS-TE and MPLS VPN Switch OpenFlow. *ACM SIGCOMM 2011 Conference on Applications, Technologies, Architectures and Protocols for Computer Communications*. Toronto, ON, Canada, August 2011.

6. Shibanov A. P., Koryachko V. P., Izhvanov Y. L. Modeling of Aggregated Telecommunication Link with Technology of Open Flows. *Radioengineering*. 2012, No. 3, P. 109–112.

7. Lemeshko A. V., Vavenko T. Y. [Improvement of Flow Model of Multipath Routing on the Basis Load Balancing]. *Problems of telecommunications*. 2012, Vol. 6, No. 1, P. 12–29 (In Russ.).

8. Lemeshko A. V., Vavenko T. V. [Development and Research of the Flow Model of Adaptive Routing in the Software-Defined Networks with Load Balancing]. *Doklady Tomskogo Gosudarstvennogo Universiteta Sistem Upravleniya i Radioyelektroniki* [Proc. of Tomsk State University of Control Systems and Radioelectronics]. 2013, Vol. 29, No. 3, P. 100–108.

9. Merindol P. Improving Load Balancing with Multipath Routing. *Proc. of the 17-th International Conference on Computer Communications and Networks (IEEE ICCCN 2008).* 2008, P. 54–61.

10. Mikhailenko V. S., Solodovnik M. S., Analysis of the Adaptive Neural Network Router. *Automatic Control and Computer Science*. 2016, Vol. 50, No. 1, P. 46–53.

11. Perepelkin D. A., Cyganov I. Ju. [Algorithm for pairwise transitions in computer networks based on the subnet routing method]. *Vestnik RGRTU*. 2016, No. 57, P. 56–62 (In Russ.).

12. Goulamghoss M. I., Bassoo V. Analysis of traffic engineering and fast reroute on multiprotocol label switching. *J Ambient Intell Human Comput.* 2021, No. 12, P. 2409–2420. https://doi.org/ 10.1007/s12652-020-02365-5.

13. Papan J., Segec P., Yeremenko O., Bridova I., Hodon M. Enhanced Multicast Repair Fast Reroute Mechanism for Smart Sensors IoT and Network Infrastructure. *Sensors*. 2020, No. 20, P. 3428. https://doi.org/10.3390/s20123428.

14. Perepelkin D. A., Byshov V. S. *Modul' dinamicheskoy balansirovki potokov dannykh v programmno-konfiguriruemykh setyakh s obespecheniem kachestva setevykh servisov* [Modul' dinamicheskoy balansirovki potokov dannyh v programmno-konfiguriruemyh setyah s obespecheniem kachestva setevyh servisov]. Patent RF, no. 2017615438, 2017.

15. Perepelkin D. A., Ivanchikova M. A. *Modul' mnogoputevoy marshrutizatsii v programmno-konfiguriruemykh setyakh na baze protokola OpenFlow* [Modul' mnogoputevoj marshrutizacii v programmno-konfiguriruemyh setjah na baze protokola OpenFlow]. Patent RF no. 2017612645, 2017.

16. Ishokov D. S., Zaripova Je. R. *Programma dinamicheskoy mezhuzlovoy balansirovki trafika* [Programma dinamicheskoj mezhuzlovoj balansirovki trafika]. Patent RF no. 2015662938, 2016.

17. Perepelkin D. A., Byshov B. S. Sistema mnogoputevoy adaptivnoy marshrutizatsii i balansirovki nagruzki v dinamicheskikh korporativnykh setyakh [Sistema mnogoputevoj adaptivnoj marshrutizacii i balansirovki nagruzki v dinamicheskih korporativnyh setjah]. Patent RF no. 2015662569, 2015.

18. Omel'chenko A. V. Teoriya grafov [Graph theory]. Moscow, MCNMO Publ., 2018, 416 p.

19. Aliev T. I. Osnovy modelirovaniya diskretnykh sistem [Fundamentals of modeling discrete systems]. SPb., SPbGU ITMO Publ., 2009, 363 p.

20. Gaipov K. Je., Krikunov I. L., Demicheva A. A. *Optimal'noe raspredelenie trafika seti massovogo obsluzhivaniya na osnove konturnogo metoda po kriteriyu minimuma poter'* [Optimal'noe raspredelenie trafika seti massovogo obsluzhivanija na osnove konturnogo metoda po kriteriju minimuma poter']. Patent RF no. 2022684654, 2022.

21. Bertsekas D., Gallager R. Seti peredachi dannykh [Data networks]. Moscow, Mir Publ., 1989, 544 p.

© Gaipov K. E., Kuznetsov A. A., Krikunov I. L., 2023

Гаипов Константин Эдуардович – кандидат технических наук, доцент кафедры электронной техники и телекоммуникаций; Сибирский государственный университет науки и технологий имени академика М. Ф. Решетнева. E-mail: gaipovke@yandex.ru.

Кузнецов Александр Алексеевич – доктор физико-математических наук, профессор, директор НОЦ «Институт космических исследований и высоких технологий»; Сибирский государственный университет науки и технологий имени академика М. Ф. Решетнева. E-mail: alex_kuznetsov80@mail.ru.

Крикунов Илья Леонидович – младший научный сотрудник НЛ «Спутниковые телекоммуникационные системы»; Сибирский государственный университет науки и технологий имени академика М. Ф. Решетнева. E-mail: zaybernev@mail.ru.

Gaipov Konstantin Eduardovich – Cand. Sc., Associate Professor of the Department of Electronic Engineering and Telecommunications; Reshetnev Siberian State University of Science and Technology. E-mail: gaipovke@yandex.ru.

Kuznetsov Alexander Alekseevich – Dr. Sc., Professor, Head of Institute of Space Research and High Technologies; Reshetnev Siberian State University of Science and Technology. E-mail: alex_kuznetsov80@mail.ru.

Krikunov Ilya Leonidovich – Junior researcher of the Scientific Library Satellite Telecommunication System; Reshetnev Siberian State University of Science and Technology. E-mail: zaybernev@mail.ru.