

Enhanced method of fast re-routing with load balancing in software-defined networks

Oleksandr Lemeshko, Oleksandra Yeremenko*

A two-level method of fast re-routing with load balancing in a software-defined network (SDN) is proposed. The novelty of the method consists, firstly, in the introduction of a two-level hierarchy of calculating the routing variables responsible for the formation of the primary and backup paths, and secondly, in ensuring a balanced load of the communication links of the network, which meets the requirements of the traffic engineering concept. The method provides implementation of link, node, path, and bandwidth protection schemes for fast re-routing in SDN. The separation in accordance with the interaction prediction principle along two hierarchical levels of the calculation functions of the primary (lower level) and backup (upper level) routes allowed to abandon the initial sufficiently large and nonlinear optimization problem by transiting to the iterative solution of linear optimization problems of half the dimension. The analysis of the proposed method confirmed its efficiency and effectiveness in terms of obtaining optimal solutions for ensuring balanced load of communication links and implementing the required network element protection schemes for fast re-routing in SDN.

Keywords: SDN, flow-based model, fast re-routing, load balancing, link utilization, primary path, backup path, calculations hierarchy

1 Introduction

One of the most promising lines of development of infocommunications is the construction of software-defined networks. The main goal of implementing SDN solutions in practice is to increase the efficiency of processes related to the provision of a given level of Quality of Service (QoS) [1-4]. Thereto, it is planned to transfer some of the basic functions for managing the network as a whole and traffic in particular from the Infrastructure Layer switching equipment to the Network Operating System (NOS) servers located at the Control Layer [5-7]. This approach, on the one hand, will make it possible to reduce the cost of using switching equipment by relieving its software, which sometimes amounts to 70 % of the cost of the network equipment. On the other hand, the removal of key management functions to the NOS level should improve the quality and centralization of the decisions made due to fuller and more adequate accounting of the network state.

One of the most important problems facing SDN is to ensure the resilience of its operation in various operating modes. In this case, depending on the location of the failed network element, this problem covers three main planes, see Fig. 1. The plane of the transmitted data (Data Plane), Control Plane, and the plane of communications between the elements of the Data and Control Planes. The problem of providing network resiliency is particularly acute at the Infrastructure Layer, due to the presence of many different types of network elements (switches, routers, gateways, communication links, etc)

and the influence of a wide range of factors that cause failures. The main of these factors traditionally are [5–9]:

- failure of the SDN equipment caused by a breakdown,
- network overload,
- compromising in terms of network security (eg malicious attacks targeted at critical software or hardware elements of the network infrastructure).

As shown by the analysis [5, 6, 9–12], fast re-routing is a sufficiently effective means of ensuring network resiliency. In the course of fast re-routing, in the case of failure of one or another network element, the transmitted packets will be almost instantaneously (not more than in 50 ms) redirected to the backup paths without waiting for the next update of the routing tables. It is important to take into account the fact that the above features of SDN construction and functioning negatively affect the scalability of protocol solutions for packet re-routing.

In this regard, the problems of re-routing in SDN undergo some modification both at the level of their statement and obtaining of the final solution. This is primarily because the calculation of the required routes is carried out on the SDN-controllers, mainly centralized by solving quite complex computationally optimization problems [13–25]. Implementation of the same principles of fault-tolerant routing is characterized by the fact that along with many primary routes it is necessary to calculate the backup paths using various schemes for protecting network elements: a link, a node or a path. This is usually accompanied, on the one hand, by an additional increase in the dimension of the routing problems; and

* Kharkiv National University of Radio Electronics, Department of Infocommunication Engineering, Kharkiv, Ukraine, 14 Nauka Ave., oleksandr.lemeshko@nure.ua, oleksandra.yeremenko.ua@ieee.org

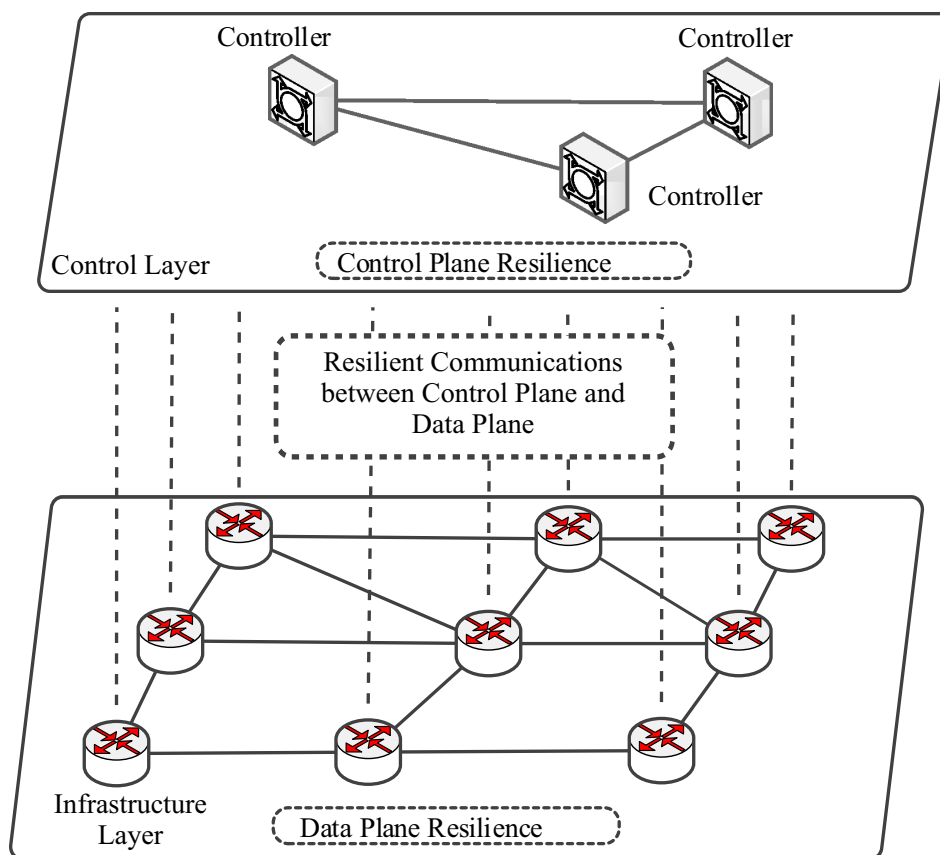


Fig. 1. Example of SDN architecture

on the other hand, by the excessive use of the network resource, especially when implementing the bandwidth protection scheme of the used paths and the network, which can negatively affect the resultant level of end-to-end Quality of Service. Therefore, during the application of fast re-routing, it is necessary to effectively use the available link and buffer resources, based, first of all, on optimal load balancing in the network [26, 27].

It is important to realize that the effectiveness of the protocol implementation and the resulting complexity of obtaining a fault-tolerant routing solution depends to a large extent on the type of mathematical model and method that form the basis for it. They, as a rule, determine the specifics of the formulation and subsequent solution of the optimization problem, to which the calculation of the required paths (primary and backup) and load balancing in the network is reduced.

2 Survey of optimization approaches for mathematical modelling of fast re-routing and load balancing processes in network

In [13–15] the task of multipath routing with load balancing under Traffic Engineering fits into the framework of the optimization problem of linear programming, which

positively affects the computational complexity of the final protocol solutions. However, for example, accounting for explicit QoS requirements when solving routing tasks [16–18] is necessarily connected with the need to solve more complex optimization problems of non-linear programming.

The transition to dynamic routing models [19–22], which is represented by differential or integral equations of state and aimed at considering the dynamics of processes occurring in the network, assumes the formulation and solution of even more computationally complex optimal control problems. In this case, not one optimal solution of the problem is calculated, but a certain trajectory of such solutions at a certain time interval, which is specified, as a rule, by a timer for recalculating (updating) the routing tables.

Special attention should be paid to the practical implementation of solutions related to fast re-routing as a subclass of fault-tolerant routing tasks. As shown by the analysis, on the one hand, the need to calculate a set of backup paths along with the primary routes implies a significant increase in the dimensionality of the formulated optimization problem [23–25]. On the other hand, the formalization of path protection schemes and their bandwidth also, as a rule, leads to a nonlinear formulation of the problem [24, 25].

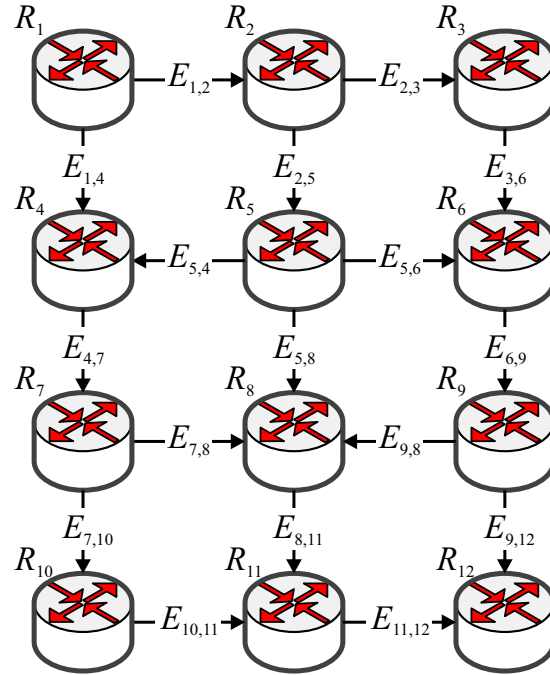


Fig. 2. Network structural model within the SDN data plane

Therefore, the scientific task, connected with the development of effective computational methods for solving problems of fast re-routing with load balancing, aimed at increasing the scalability of their application in software-defined networks, is relevant.

3 Flow-based model of fast re-routing in SDN

3.1 Network structural model

Suppose that the structure of the SDN Data Plane is described by the graph $G = (R, E)$ in Fig. 2, where $R = \{R_i; i = 1, \dots, m\}$ is the set of routers in the network, and $E = \{E_{i,j}; i, j = 1, \dots, m, i \neq j\}$ is the set of links.

Let us denote by $R_i^* = \{R_j^*; E_{j,i} \neq 0, j = 1, \dots, m, i \neq j\}$ the subset of routers adjacent to the R_i router. Then the number of communication links in the network is defined by $n = |E|$, and to each of links the capacity $\varphi_{i,j}$ will be assigned.

3.2 Functional model of fast re-routing in SDN

When solving the fast re-routing problem in SDN, we use the functional model described in [23–25]. It belongs to the class of flow-based solutions and focuses on the optimal use of available network resources during the process of routing in network. Within the framework of this model, with each unicast k th flow, a number of functional parameters are associated: s_k is the source router; d_k is the destination router; λ^k is the average intensity

of packets of the k th flow measured in packets per second (1/s). Let K be the set of flows transmitted in SDN, then $k \in K$.

The result of solving the problem of fast re-routing is calculation on the NOS server of two types of routing variables $x_{i,j}^k$ and $\bar{x}_{i,j}^k$ that characterize the proportion of the intensity of the k th flow within the link $E_{i,j} \in E$, which is included in the primary or backup path respectively. The total number of routing variables responsible for the formation of both primary and backup routes depends on the number of flows and communication links in the network as follows: $2|K||E|$.

In the case when SDN, for example, uses single path strategy for routing variables of two types, the following constraints take place

$$x_{i,j}^k \in \{0; 1\} \quad \text{and} \quad \bar{x}_{i,j}^k \in \{0; 1\}. \quad (1)$$

To ensure the connectivity of the calculated routes, the conditions for conservation the flow are introduced separately for the routing variables of the primary path [14, 15]

$$\begin{aligned} \sum_{j: E_{i,j} \in E} x_{i,j}^k - \sum_{j: E_{j,i} \in E} x_{j,i}^k &= 0; \quad k \in K, \quad R_i \neq s_k, d_k; \\ \sum_{j: E_{i,j} \in E} x_{i,j}^k - \sum_{j: E_{j,i} \in E} x_{j,i}^k &= 1; \quad k \in K, \quad R_i = s_k; \\ \sum_{j: E_{i,j} \in E} x_{i,j}^k - \sum_{j: E_{j,i} \in E} x_{j,i}^k &= -1; \quad k \in K, \quad R_i = d_k; \end{aligned} \quad (2)$$

and for the backup path routing variables [24, 25]

$$\begin{aligned} \sum_{j: E_{i,j} \in E} \bar{x}_{i,j}^k - \sum_{j: E_{j,i} \in E} \bar{x}_{j,i}^k &= 0; \quad k \in K, R_i \neq s_k, d_k; \\ \sum_{j: E_{i,j} \in E} \bar{x}_{i,j}^k - \sum_{j: E_{j,i} \in E} \bar{x}_{j,i}^k &= 1; \quad k \in K, R_i = s_k; \\ \sum_{j: E_{i,j} \in E} \bar{x}_{i,j}^k - \sum_{j: E_{j,i} \in E} \bar{x}_{j,i}^k &= -1; \quad k \in K, R_i = d_k. \end{aligned} \quad (3)$$

3.3 Conditions of network elements protection with fast re-routing in SDN

As shown by the analysis [10–12, 24, 25], in the course of fast re-routing, several basic schemes for protecting network elements can be supported: node, link, path and its bandwidth. In [23, 25], the mathematical conditions were obtained to support the mentioned above protection schemes as components of the corresponding mathematical models.

In [25], when implementing a protection scheme for the link $E_{i,j} \in E$, it is proposed to set the additional limitations, similar to (1), to routing variables $\bar{x}_{i,j}^k$, responsible for determination of a backup path. In the case of using the single path routing over the backup paths, the condition of the form is introduced

$$\bar{x}_{i,j}^k \in \{0; \delta_{i,j}^k\}, \quad (4)$$

$$\delta_{i,j}^k = \begin{cases} 0, & \text{when protecting the link } E_{i,j}; \\ 1, & \text{otherwise.} \end{cases} \quad (5)$$

Fulfilling the conditions (4) and (5) ensures that the protected link $E_{i,j} \in E$ will not use a backup route in the implementation of a single path routing. The conditions (4) and (5) are linear in contrast to the nonlinear solutions proposed in [24], thereby reducing the computational complexity of producing the final protocol solutions.

With the implementation of the node $R_i \in R$ protection scheme the conditions (4) and (5) are generalized for the case when there is a need to protect a set of links incident to the protected node [25]. Then, when implementing the single path routing, the following limitations are taking place

$$\bar{x}_{i,j}^k \in \{0; \delta_{i,j}^k\} \text{ under } R_j \in R_i^*, j = 1, \dots, m. \quad (6)$$

where selection of values $\delta_{i,j}^k$ is subject to condition (5).

Thus, meeting the requirements of (6) guarantees the protection of the node $R_i \in R$ by banning backup route to use all links that come out of this node. Since only transit routers should be protected, the ban on the use of outgoing links under the conditions (3) prevents inclusion of incoming links into the backup path for this node, and as a result, this helps to protect the node R_i as a whole.

It is worth noting that the protection conditions of pre-determined nodes and network links are, as a rule, linear,

and their accounting does not critically affect the calculation of routing variables $x_{i,j}^k$ and $\bar{x}_{i,j}^k$ responsible for the formation of a set of primary and backup routes. There is a more ambiguous situation with the formalization of the conditions for the protection of the path and its bandwidth. In [25], it is proposed to introduce the following conditions into the routing model to ensure protection of the calculated primary path for each k th flow

$$\sum_{R_j \in R_i^*} \sum_{R_p \in R_i^*} x_{j,i}^k \bar{x}_{p,i}^k = 0, \quad \forall R_i \in R \setminus \{s_k, d_k\}. \quad (7)$$

Bilinear conditions (7) are recorded for each transit router, ie for all routers, except the source and the destination. The physical meaning of these conditions is that none of the network routers is used by the primary and backup routes simultaneously. This will prevent the intersection of the primary and backup routes by both nodes and links.

In solving problems of single path routing (1), the following conditions for protecting the bandwidth of calculated paths are proposed in [24]

$$\sum_{k \in K} \lambda^k \frac{x_{i,j}^k + \bar{x}_{i,j}^k}{x_{i,j}^k \bar{x}_{i,j}^k + 1} \leq \varphi_{i,j}, \quad E_{i,j} \in E. \quad (8)$$

The fulfillment of (8) ensures that the SDN communication links will not be overloaded with the flows transmitted by them, even if they are fast rerouted. The conditions (8) introduce functional redundancy into the network, because the bandwidth of the communication links is allocated both for the primary and backup routes, to which a particular k th flow can hypothetically switch.

From the mathematical point of view, conditions (7) and (8) are nonlinear, which negatively affects the computational complexity of obtaining finite solutions in relation to fast re-routing in SDN. Thus, in the course of the analysis it was established that the main factors affecting the scalability and computational complexity of the centralized solution of the fast re-routing problems are:

- increased (actually doubled) dimension of the problem ($2|K||E|$), because the routing variables (1) responsible for the formation of both primary and backup routes are subject to calculation;
- bilinear nature of path protection conditions (7);
- nonlinear character of the conditions for protecting the bandwidth of the calculated primary routes in SDN (8).

In this context, the fast re-routing method proposed in this paper should minimize the noted shortcomings of the model (1)–(8) when implemented in software-defined networks.

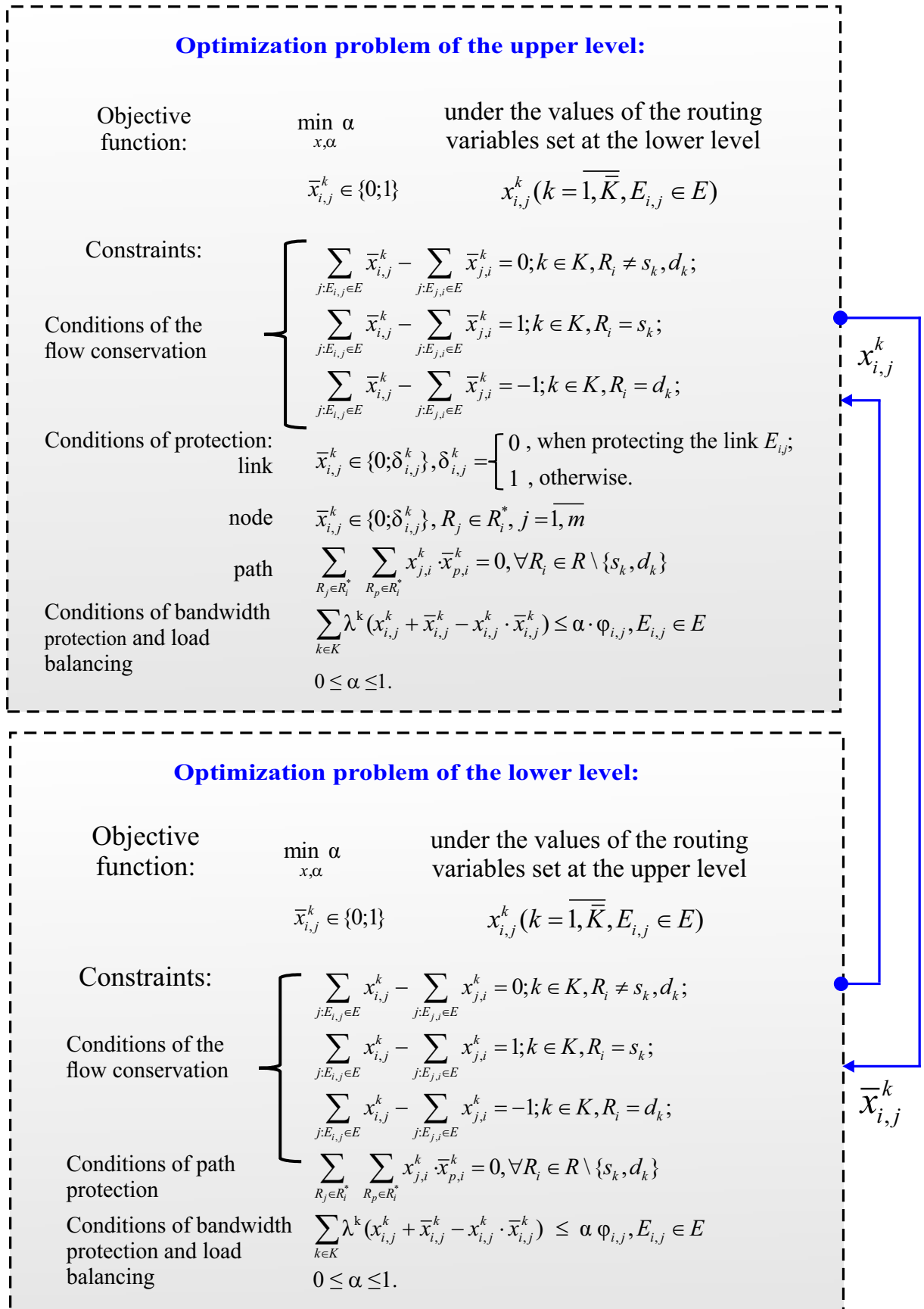


Fig. 3. The structure of the method of fast re-routing with load balancing in software-defined networks

4 Method of fast re-routing with load balancing in SDN

In order to increase the scalability of solutions within the framework of the method of fast re-routing in SDN, it is proposed:

- to change the form of (8) with the preservation of their physical meaning;
- to form the problem of calculating routing variables $x_{i,j}^k$ and $\bar{x}_{i,j}^k$ as optimization, related to load balancing in SDN;
- to introduce a two-level functional hierarchy for calculating routing variables of different types.

Conditions (8) introduced in [24] are proposed in the form

$$\sum_{k \in K} \lambda^k (x_{i,j}^k + \bar{x}_{i,j}^k - x_{i,j}^k \bar{x}_{i,j}^k) \leq \alpha \varphi_{i,j}, \quad E_{i,j} \in E \quad (9)$$

where α is the additional control variable that numerically determines the upper bound of the SDN communication links utilization and obeys the following conditions [13–15]

$$0 \leq \alpha \leq 1. \quad (10)$$

It should be noted that the left-hand sides of inequalities (8) and (9) give the same numerical result.

An optimality criterion of the problem solutions of the fast re-routing in SDN, by analogy with the results obtained in [13–15], will be the minimum of the bound introduced in (9), that is

$$\min_{x, \bar{x}, \alpha} \alpha. \quad (11)$$

The fulfillment of conditions (9)–(11) helps to ensure optimal load balancing when implementing the scheme for protecting the bandwidth of calculated paths.

Within the framework of the proposed method, the two-level hierarchy of calculations introduced (see Fig. 3) is subject to the interaction prediction principle of the theory of hierarchical multi-level systems [28, 29]. Then, at the lower level, it is proposed to calculate the routing variables $x_{i,j}^k$ responsible for determining the primary paths in the network, while minimizing the bound α (11), but for fixed values $\bar{x}_{i,j}^k$ set at the upper hierarchical level. It is important to satisfy the constraint conditions (1), (2), (9), (10), and when the path is protected - to additionally satisfy conditions (7). With this formulation of the problem, (7) and (9) will already be linear, since the values $\bar{x}_{i,j}^k$ at the lower level are known.

At the upper level of the hierarchy, the calculation of the routing variables $\bar{x}_{i,j}^k$ responsible for the formation (and actually predicting) of the backup routes is also performed by minimizing α (11) with fixed and set at the lower level values of the routing variables $x_{i,j}^k$. In

the course of optimization, it is necessary to fulfill the conditions:

- (1), (3), (7), (9), and (10), when implementing the path protection scheme;
- (1), (3), (9), (10), and (4) or (6), when implementing link or node protection schemes.

Since for the upper level of calculation the values $x_{i,j}^k$ are known, conditions (7) and (9) also become linear.

Thus, the process of solving the formulated problem of fast re-routing in SDN acquires an iterative character. At the same time, the criterion for the completion of calculations is the achievement of the optimum (11), which reveals itself in the proximity of the values of the objective function (11) calculated at adjacent iterations, but at different hierarchical levels.

It is important to note that with the protection of the link, node, path, and bandwidth in the network, the application of the described method allows us to abandon the solution of the original nonlinear and sufficiently large optimization problem by transiting to the iterative solution of linear optimization problems of half the dimension (Fig. 3). This certainly affects the overall scalability of the fast re-routing solutions with load balancing in the network as a whole.

Then the effectiveness of the proposed method directly depends on the number of iterations for which it provides a search for the optimal values from the point of view of criterion (11) of the values of the routing variables (1) responsible for the formation of the primary and backup routes. This functional feature of the proposed method is subject to additional investigation.

Table 1. Capacities of the network links

Link	Capacity	Link	Capacity	Link	Capacity
$E_{1,2}$	800	$E_{5,6}$	300	$E_{7,10}$	500
$E_{2,3}$	500	$E_{4,7}$	700	$E_{8,11}$	900
$E_{1,4}$	800	$E_{5,8}$	500	$E_{9,12}$	800
$E_{2,5}$	900	$E_{6,9}$	800	$E_{10,11}$	700
$E_{3,6}$	700	$E_{7,8}$	400	$E_{11,12}$	600
$E_{5,4}$	400	$E_{9,8}$	500		

5 Analysis of the proposed method of fast re-routing with load balancing in SDN

The analysis of the proposed fast re-routing method was carried out on a number of network configurations and for a different number of flows and their characteristics. The features of the method work will be demonstrated on a numerical example. The structure of the network is shown in Fig. 2. In Table 1, the capacities of communication links of the network are indicated.

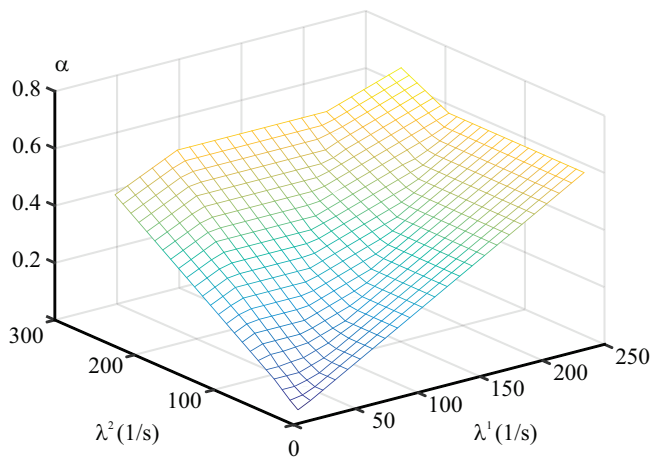


Fig. 4. Dependence of the network links utilization bound (10) on the intensities of the transmitted packet flows (when protecting paths for two unicast flows)

Suppose in the network according to Fig. 2, it is necessary to provide the solution of the tasks of fast re-routing with the implementation of the path protection scheme for two unicast flows, the characteristics of which are given in Table 2. The intensity of both flows varies from 50 to 250 1/s.

Table 2. Flow Characteristics

Flow Number	Flow intensity	Source
1	$\lambda^1 = 50$ to 250 1/s	R_1
2	$\lambda^2 = 50$ to 250 1/s	R_5

As the results of the analysis have showed, the method of fast re-routing with load balancing in SDN, proposed in Section 4 for the initial data presented in Table 1, provided finding the optimal values of the routing variables (1) and the links utilization bound (10) (Fig. 4) on average for 2-3 iterations (Fig. 5). In addition, as can be seen from Fig. 4 and Fig. 5, the number of iterations increases to three in the case of equal or close values of the intensities of the transmitted flows. In Table 3, for example, the results of calculations for the three iterations of the method work at $\lambda^1 = 240$ 1/s and $\lambda^1 = 220$ 1/s are shown. In this table, the utilization for each link $E_{i,j} \in E$ is indicated

$$\alpha_{i,j} = \frac{\sum_{k \in K} x_{i,j}^k \lambda^k}{\varphi_{i,j}}. \quad (12)$$

In fact, α (11) is a threshold (an upper bound), that is, the maximum value among the set of coefficients $\alpha_{i,j}$ (12).

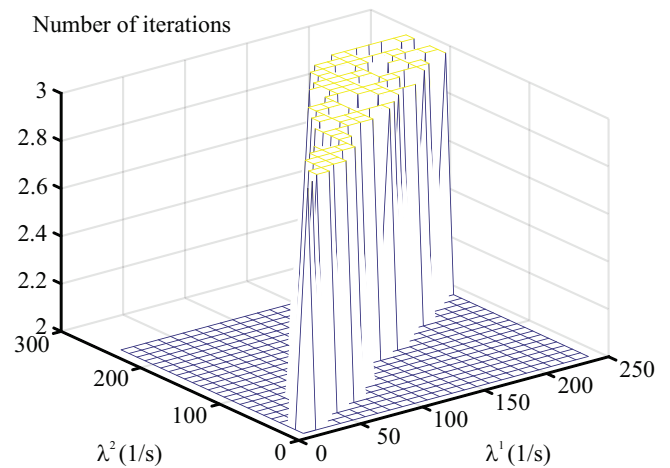


Fig. 5. Dependence of the number of iterations of the proposed method on the intensities of the transmitted packet flows (when protecting paths for two unicast flows)

After the first iteration of the method, the utilization of the link $E_{7,10}$ determines the value of the utilization bound of the network communication links and is 0.92 (Table 3). In this case, for the first flow, the primary path passes through the routers $R_1 \rightarrow R_4 \rightarrow R_7 \rightarrow R_{10} \rightarrow R_{11} \rightarrow R_{12}$, and the backup path $R_1 \rightarrow R_2 \rightarrow R_5 \rightarrow R_6 \rightarrow R_9 \rightarrow R_{12}$. While for the second flow, the primary path is determined by routers $R_5 \rightarrow R_8 \rightarrow R_{11}$, and the backup path $R_5 \rightarrow R_4 \rightarrow R_7 \rightarrow R_{10} \rightarrow R_{11}$.

After the second iteration, the method provides a further reduction in the load bound of the communication links to 0.8, which in this case (Table 3) is determined by the link $E_{5,6}$ utilization. Here, the primary path for the first flow is $R_1 \rightarrow R_4 \rightarrow R_7 \rightarrow R_8 \rightarrow R_{11} \rightarrow R_{12}$, and the backup path $R_1 \rightarrow R_2 \rightarrow R_5 \rightarrow R_6 \rightarrow R_9 \rightarrow R_{12}$. In this case, for the second flow, the primary and backup paths remain the same.

Only after the third iteration, the method determined the final solution of the problem of fast re-routing with $\alpha = 0.657$ (Table 3). Here, for the first flow, the primary path is represented by routers $R_1 \rightarrow R_4 \rightarrow R_7 \rightarrow R_8 \rightarrow R_{11} \rightarrow R_{12}$, and the backup path $R_1 \rightarrow R_2 \rightarrow R_3 \rightarrow R_6 \rightarrow R_9 \rightarrow R_{12}$. As before, for the second flow, the primary and backup paths remained unchanged.

Let us demonstrate the features of the proposed method when implementing the link $E_{8,11}$ protection scheme for two unicast flows, the characteristics of which are given in Table 2. In this case, in accordance with the conditions (4), link $E_{8,11}$ protection actually implies the protection of the router R_8 . As the results of the analysis showed, the method of fast re-routing in SDN, proposed in Section 4 for the initial data presented in Table 1, ensured finding the optimal values of the routing variables (1) and the links utilization bound (10) (Fig. 6) on average also for 2-3 iterations (Fig. 7).

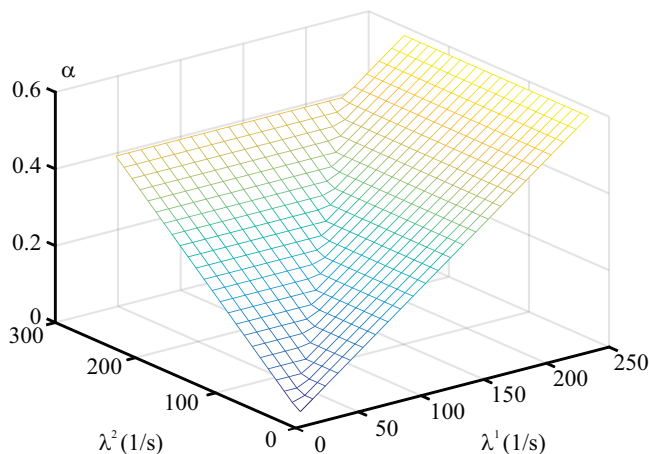


Fig. 6. Dependence of the network links utilization bound (10) on the intensities of the transmitted packet flows (when protecting the link $E_{8,11}$)

The proposed two-level method of fast re-routing, as in the implementation of the path protection scheme (Fig. 4), provided a smooth enough change in the utilization bound of the network communication links (10). This is the advantage of the resulting solution, because it contributes to the corresponding smooth change of key quality of service indicators (average delay and probability of packet loss), which largely depend on the utilization of network links.

In Table 4, for example, the results of calculations for two iterations of the method at $\lambda^1 = 240$ 1/s and $\lambda^2 = 240$ 1/s are shown. After the first iteration of the method, the utilization of the link $E_{7,10}$ determines the value of the utilization bound of the network communication links and is 0.96 (Table 4). In this case, for the first flow, the primary path passes through the routers $R_1 \rightarrow R_4 \rightarrow R_7 \rightarrow R_{10} \rightarrow R_{11} \rightarrow R_{12}$, and the backup path $R_1 \rightarrow R_2 \rightarrow R_5 \rightarrow R_6 \rightarrow R_9 \rightarrow R_{12}$. Whereas for the second flow, the primary path passes through the routers $R_5 \rightarrow R_8 \rightarrow R_{11}$, and the backup path is $R_5 \rightarrow R_4 \rightarrow R_7 \rightarrow R_{10} \rightarrow R_{11}$.

After the second iteration, the method provided a further reduction in the utilization bound of the communication links to 0.8, which in this case (Table 4) is determined by the link $E_{5,6}$ utilization. Here, for the first flow, the primary path contains routers $R_1 \rightarrow R_2 \rightarrow R_3 \rightarrow R_6 \rightarrow R_9 \rightarrow R_{12}$, and the backup path $R_1 \rightarrow R_2 \rightarrow R_5 \rightarrow R_6 \rightarrow R_9 \rightarrow R_{12}$. While for the second flow, the primary and backup paths coincide and take the form $R_5 \rightarrow R_4 \rightarrow R_7 \rightarrow R_{10} \rightarrow R_{11}$, but no longer contain the protected link $E_{8,11}$.

The proposed method calculated the optimal order of fast re-routing with load balancing of two packet flows after the third iteration with $\alpha = 0.6$ (Table 4). This utilization can be observed for the link $E_{5,4}$. In this case, for the first flow, the primary and backup paths coincide and pass through the routers $R_1 \rightarrow R_2 \rightarrow R_3 \rightarrow R_6 \rightarrow$

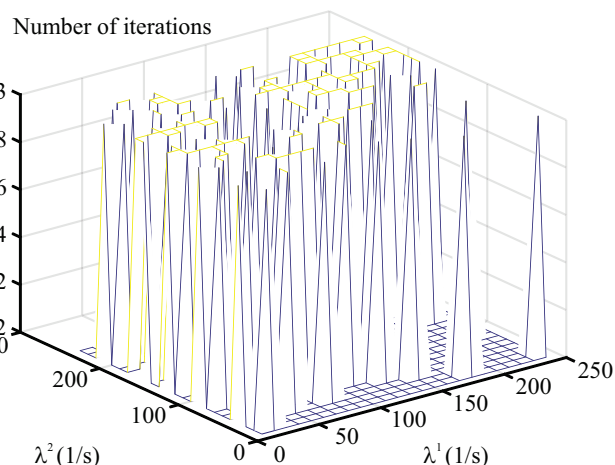


Fig. 7. Dependence of the number of iterations of the proposed method on the intensities of the transmitted packet flows (when protecting the link $E_{8,11}$)

$R_9 \rightarrow R_{12}$, as well as for the second flow routers $R_5 \rightarrow R_4 \rightarrow R_7 \rightarrow R_{10} \rightarrow R_{11}$ define both the primary and backup paths. However, none of the calculated routes contains a protected network element - link $E_{8,11}$.

As shown in Table 4, the primary paths calculated for each of the transmitted flows can pass through the protected elements of the network - the link $E_{8,11}$ and the router R_6 . Backup paths did not contain protected links and network routers.

The above example demonstrated the operability of the proposed two-level method of fast re-routing with load balancing. In this case, the final value of the utilization bound of the network communication links (11), obtained using the proposed method, fully corresponded to the results of the centralized calculation of the routing variables $x_{i,j}^k$ and $\bar{x}_{i,j}^k$ using the criterion (11) and the presence of constraints (1)–(10). The calculation of the control variables (1), (10) in solving the mixed integer linear programming (MILP) optimization problem represented by the objective function (11) have been performed using Optimization Toolbox of the MatLab environment, namely `intlinprog` function. In this case according to (9) the solution of two problems was obtained, which are fast re-routing and load balancing.

The results of the research for other network structures and initial data also allowed to formulate the conditions under which the proposed two-level method demonstrated the maximum efficiency from the point of view of obtaining optimal solutions for providing fast re-routing with load balancing of SDN links by criterion (11) and implementing the required network elements protection schemes. First of all, it should be attributed to the fact that as the size of the network structures and the connectivity of the SDN routers increase due to the presence of a larger number of routes between the source-destination pairs and the additional available network (link) resource,

Table 3. Calculation results after the first, second, and third method iterations with path protection scheme

Link	First iteration, $\alpha = 0.92$					Second iteration, $\alpha = 0.8$					Third iteration, $\alpha = 0.657$							
	First flow		Second flow			$\alpha_{i,j}$	First flow		Second flow			$\alpha_{i,j}$	First flow		Second flow			$\alpha_{i,j}$
	Path				Path				Path									
P	B	P	B		P	B	P	B		P	B	P	B					
$E_{1,2}$	0	240	0	0	0.3	0	240	0	0	0.3	0	240	240	0	0.3			
$E_{2,3}$	0	0	0	0	0	0	0	0	0	0	0	240	240	0	0.48			
$E_{1,4}$	240	0	0	0	0.3	240	0	0	0	0.3	240	0	0	0	0.3			
$E_{2,5}$	0	240	0	0	0.267	0	240	0	0	0.267	0	240	0	0	0			
$E_{3,6}$	0	0	0	0	0	0	0	0	0	0	0	0	240	0	0.343			
$E_{5,4}$	0	0	0	220	0.55	0	0	0	220	0.55	0	0	0	220	0.55			
$E_{5,6}$	0	240	0	0	0.8	0	240	0	0	0.8	0	240	0	0	0			
$E_{4,7}$	240	0	0	220	0.657	240	0	0	220	0.657	240	0	0	220	0.657			
$E_{5,8}$	0	0	220	0	0.44	0	0	220	0	0.44	0	0	0	0	0.44			
$E_{6,9}$	0	240	0	0	0.3	0	240	0	0	0.3	0	240	240	0	0.3			
$E_{7,8}$	0	0	0	0	0	240	0	0	0	0.6	240	0	0	0	0.6			
$E_{9,8}$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
$E_{7,10}$	240	0	0	220	0.92	0	0	0	220	0.44	0	0	0	220	0.44			
$E_{8,11}$	0	0	220	0	0.244	240	0	220	0	0.511	240	0	0	0	0.511			
$E_{9,12}$	0	240	0	0	0.3	0	240	0	0	0.3	0	240	240	0	0.3			
$E_{10,11}$	240	0	0	220	0.657	0	0	0	220	0.314	0	0	0	220	0.314			
$E_{11,12}$	240	0	0	0	0.4	240	0	0	0	0.4	240	0	0	0	0.4			

P = primary, B= Backup

Table 4. Calculation results after the first, second, and third method iterations when protecting the link $E_{8,11}$

Link	First iteration, $\alpha = 0.96$					Second iteration, $\alpha = 0.8$					Third iteration, $\alpha = 0.6$							
	First flow		Second flow			$\alpha_{i,j}$	First flow		Second flow			$\alpha_{i,j}$	First flow		Second flow			$\alpha_{i,j}$
	Path				Path				Path									
P	B	P	B		P	B	P	B		P	B	P	B					
$E_{1,2}$	0	240	0	0	0.3	240	240	0	0	0.3	240	240	0	0	0.3			
$E_{2,3}$	0	0	0	0	0	240	0	0	0	0.48	240	240	0	0	0.48			
$E_{1,4}$	240	0	0	0	0.3	0	0	0	0	0	0	0	0	0	0			
$E_{2,5}$	0	240	0	0	0.267	0	240	0	0	0.267	0	0	0	0	0			
$E_{3,6}$	0	0	0	0	0	240	0	0	0	0.323	240	240	0	0	0.343			
$E_{5,4}$	0	0	0	240	0.6	0	0	240	240	0.6	0	0	240	240	0.6			
$E_{5,6}$	0	240	0	0	0.8	0	240	0	0	0.8	0	0	0	0	0			
$E_{4,7}$	240	0	0	240	0.686	0	0	240	240	0.343	0	0	240	240	0.343			
$E_{5,8}$	0	0	240	0	0.48	0	0	0	0	0	0	0	0	0	0			
$E_{6,9}$	0	240	0	0	0.3	240	240	0	0	0.3	240	240	0	0	0.3			
$E_{7,8}$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
$E_{9,8}$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
$E_{7,10}$	240	0	0	240	0.96	0	0	240	240	0.48	0	0	240	240	0.48			
$E_{8,11}$	0	0	240	0	0.267	0	0	0	0	0	0	0	0	0	0			
$E_{9,12}$	0	240	0	0	0.3	240	240	0	0	0.3	240	240	0	0	0.3			
$E_{10,11}$	240	0	0	240	0.686	0	0	240	240	0.343	0	0	240	240	0.343			
$E_{11,12}$	240	0	0	0	0.4	0	0	0	0	0	0	0	0	0	0			

P = primary, B= Backup

the proposed method makes it possible to provide better load balancing by the criterion (11) in solving problems of fast re-routing. This is especially noticeable in the implementation of path and bandwidth protection schemes, which require an excess network resource.

In addition, the proposed method demonstrated the maximum efficiency with fast re-routing of a large number of flows with different sources and destinations, as well as packet intensities. These features also positively affect the load balancing process in the implementation both single path and multipath routing.

6 Conclusion

In conventional networks, for example, in IP networks, distributed routing is used, which defines well-scalable solutions. To increase the efficiency of telecommunications solutions, in practice, Software-Defined Networks are increasingly being used, in which some of the key functions for traffic management are centralized and transferred to controllers of the network operating system, which, however, may negatively affect their scalability. In this regard, it is necessary to scalably adapt the SDN architecture to the theoretical and protocol solutions for such important traffic management tasks as fast re-routing and load balancing that meets the requirements of the traffic engineering concept.

Thus, the paper proposes the two-level method of fast re-routing with load balancing in a software-defined networks, which provides implementation of link, node, path, and bandwidth protection schemes. It is shown that effective implementation of the fast re-routing with load balancing principles in the SDN architecture can be based on the use of the mathematical model (1)–(8), which assumes the centralization of the solution of rather complicated routing tasks on the corresponding SDN-controllers. In this case, the complexity factors are their high dimensionality ($2|K||E|$) and nonlinearity (7), (8), which implies increased requirements for the performance of these controllers.

Therefore, within the presented calculation method, in accordance with the interaction prediction principle, it is proposed to divide the hierarchical levels of the functions for calculating the primary (lower level) and backup (upper level) routes along with a special recording of previously known conditions for protecting the network bandwidth (9). This made it possible to abandon the initial sufficiently large and nonlinear optimization problem by transiting to the iterative solution of linear optimization problems of half the dimension. Implementation of this approach in practice will significantly reduce the requirements for computing power, and hence the cost of SDN-controllers, which are entrusted with centralized solutions for routing in the network. At the same time, the efficiency of solutions obtained through the proposed two-level method, estimated by the upper bound of the network communication links utilization (10), (11), fully corresponded to the results of centralized calculations. This

indicates the effectiveness and adequacy of the proposed solution.

The analysis of the proposed two-level method (Figs. 4–7), confirmed its efficiency and effectiveness in terms of obtaining optimal solutions for ensuring balanced load of communication links and implementing the required network element protection schemes for fast re-routing in SDN.

REFERENCES

- [1] R. White and J. E. Tantsura, *Navigating Network Complexity: Next-Generation Routing with SDN, Service Virtualization, and Service Chaining*, Addison-Wesley Professional, 2015.
- [2] A. S. Monge and K. G. Szarkowicz, *MPLS the SDN Era: Interoperable Scenarios to Make Networks Scale to New Services*, O'Reilly Media, 2016.
- [3] T. D. Nadeau and K. Gray, *SDN, Software Defined Networks: An Authoritative Review of Network Programmability Technologies*, O'Reilly Media, 2013.
- [4] G. Pujolle, *Software Networks: Virtualization, SDN, 5G, Security*, John Wiley & Sons, 2015.
- [5] J. Rak, D. Papadimitriou, H. Niedermayer and P. Romero, "Information-Driven Network Resilience: Research Challenges and Perspectives", *Optical Switching and Networking*, vol. 23, Part 2, pp. 156–178, January 2017.
- [6] J. Domzal and R. Wojcik, "Efficient and Reliable Transmission Flow-Aware Networks - an Integrated Approach based on SDN Concept", *Proc. 2017 International Conference on Computing Networking and Communications (ICNC)*, 26-29 January 2017, pp. 837–842.
- [7] J. Domzal, Z. Dulinski, J. Rozsa, P. Gawowicz, E. Biernacka and R. Wojcik, "Automatic Hidden Bypasses Software-Defined Networks", *Journal of Network and Systems Management*, vol. 25, no. 3, pp. 457–480, 1 July 2017.
- [8] P. Smith, D. Hutchison, J. P. Sterbenz, M. Schöller, A. Fessi, M. Karaliopoulos, C. Lac and B. Plattner, "Network Resilience: a Systematic Approach", *IEEE Communications Magazine*, vol. 49, no. 7, pp. 88–97, July 2011.
- [9] D. Tipper, "Resilient Network Design: Challenges and Future Directions", *Telecommunication Systems*, vol. 56, no. 1, pp. 5–16, May 2014.
- [10] I. Hussain, *Fault-Tolerant IP and MPLS Networks (Networking Technology)*, Indianapolis: Cisco Press, 2005.
- [11] I. Koren and C. Krishna, *Fault-Tolerant Systems*, Morgan Kaufmann, 2007.
- [12] R. Kuhn, B. Hanafee, J. Allen, *Reactive Design Patterns*, Manning Publications, 2017.
- [13] Y. Seok, Y. Lee, Y. Choi and C. Kim, "Dynamic Constrained Traffic Engineering for Multicast Routing", *Proc. Information Networking: Wired Communications and Management*, vol. 2343, September 2002, pp. 278–288.
- [14] Y. Wang and Z. Wang, "Explicit Routing Algorithms for Internet Traffic Engineering", *Proceedings Eight International Conference on Computer Communications and Networks (Cat. no. 99EX370)*, 11-13 Oct.1999, pp. 582–588.
- [15] Y. Seok, Y. Lee, C. Kim and Y. Choi, "Dynamic Constrained Multipath Routing for MPLS Networks", *Proceedings Tenth International Conference on Computer Communications and Networks (Cat. no. 01EX495)*, 15-17 Oct.2001, pp. 348–353.
- [16] O. V. Lemesheko, S. V. Garkusha, O. S. Yeremenko and A. M. Hailan, "Policy-based QoS Management Model for Multiservice Networks", *Proc. 2015 International Siberian Conference on Control and Communications (SIBCON)*, 21-23 May 2015, pp. 1–4.

- [17] A. V. Lemeshko, O. Yu. Evseeva and S. V. Garkusha, "Research on Tensor Model of Multipath Routing Telecommunication Network with Support of Service Quality by Greater Number of Indices", *Telecommunications and RadioEngineering*, vol. 73, no. 15, pp. 1339–1360, 2014.
- [18] O. S. Yeremenko, O. V. Lemeshko, O. S. Nevzorova and A. M. Hailan, "Method of Hierarchical QoS Routing Based on the Network Resource Reservation", *Proc. 2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON)*, May 29 - June 2, 2017, pp. 971–976.
- [19] K. Xu, D. Tipper, Y. Qian, P. Krishnamurthy and S. Tipmongkongsilp, "Time-Varying Performance Analysis of Multihop Wireless Networks with CBR Traffic", *IEEE Transactions on Vehicular Technology*, vol. 63, no. 7, pp. 3397–3409, September 2014.
- [20] K. A. Alnowibet, "Nonstationary Erlang Loss Queues and Networks", *PhD Thesis*, North Carolina State University, 2004.
- [21] O. V. Lemeshko and O. S. Yeremenko, "Dynamics Analysis of Multipath QoS-Routing Tensor Model with Support of Different Flows Classes", *Proc. 2016 International Conference on Smart Systems and Technologies (SST)*, 12-14 Oct 2016, pp. 225–230.
- [22] O. V. Lemeshko, O. S. Yeremenko and A. M. Hailan, "QoS Solution of Traffic Management based on the Dynamic Tensor Model the Coordinate System of Interpolar Paths and Internal Node Pairs", *Proc. 2016 International Conference Radio Electronics & Info Communications (UkrMiCo)*, 11-16 September 2016, pp. 1-6.
- [23] T. Gomes, L. Martins, S. Ferreira, M. Pascoal and D. Tipper, "Algorithms for Determining a Node-Disjoint Path Pair Visiting Specified Nodes", *Optical Switching and Networking*, vol. 23, Part 2, pp. 189–204, January 2017.
- [24] O. Lemeshko, K. Arous and N. Tariki, "Effective Solution for Scalability and Productivity Improvement Fault-Tolerant Routing", *Proc. 2015 Second International Scientific-Practical Conference Problems of Infocommunications Science and Technology (PIC S&T)*, 13-15 October 2015, pp. 76–78.
- [25] O. S. Yeremenko, O. V. Lemeshko and N. Tariki, "Fast ReRoute Scalable Solution with Protection Schemes of Network Elements", *Proc. 2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON)*, May 29 – June 2, 2017, pp. 783–788.
- [26] M. Luo, Y. Zeng, J. Li and W. Chou, "An Adaptive Multi-Path Computation Framework for Centrally Controlled Networks", *Computer Networks*, vol. 83, no. C, pp. 30–44, June 2015.
- [27] S. C. Lin, P. Wang and M. Luo, "Control Traffic Balancing Software Defined Networks", *Computer Networks*, vol. 106, pp. 260–271, September 2016.
- [28] M. D. Mesarovic, D. Macko and Y. Takahara, *Theory of Hierarchical, Multilevel, System*, Academic Press, 1970.
- [29] M. G. Singh and A. Titli, *Systems: Decomposition, Optimization and Control*, Oxford: Pergamon, 1978.

Received 30 May 2017

Oleksandr Lemeshko born in 1972, received his PhD degree in Weapons and Military Equipment from the Kharkiv Institute of the Air Force, Ukraine (1999), DSc degree in Telecommunication Systems and Networks from the Kharkiv National University of Radio Electronics, Ukraine (2005), and academic rank of Professor (2012). He joined the Department of Telecommunication Systems (currently Department of Infocommunication Engineering) at the Kharkiv National University of Radio Electronics as an associate professor in 2005. He has been a professor of the Department of Telecommunication Systems since 2006. His current research interests are Traffic Management, Optimization in Telecommunications, Wireless Mesh Networks, Routing, Traffic Engineering, QoS, QoE, Hierarchical Routing, and Fault-Tolerant Routing.

Oleksandra Yeremenko received her PhD in Telecommunication Systems and Networks from the Kharkiv National University of Radio Electronics, Ukraine (2008) and academic rank of Senior Researcher (2012), Associate Professor (2017). She joined the Department of Telecommunication Systems (currently Department of Infocommunication Engineering) at the Kharkiv National University of Radio Electronics as a senior research assistant in 2007. She has been an associate professor of the Department of Telecommunication Systems since 2011. Her current research interests are Future Internet, Quality of Service, Communication Network Resilience, Flow-based Networking, Fault-Tolerant Routing, Software-Defined Networking, and Network Security.