A Revised Gomory-Hu Algorithm Taking Account of Physical Unavailability of Network Channels

Winfried Auzinger[1](✉), Kvitoslava Obelovska[2], and Roksolyana Stolyarchuk[3]

[1] Institute of Analysis and Scientific Computing, TU Wien, Vienna, Austria
winfried.auzinger@tuwien.ac.at

[2] Institute of Computer Science and Information Technologies, Lviv Polytechnic National University, Lviv, Ukraine
obelyovska@gmail.com

[3] Institute of Applied Mathematics and Fundamental Sciences, Lviv Polytechnic National University, Lviv, Ukraine
sroksolyana@yahoo.com

http://www.asc.tuwien.ac.at/~winfried/

Abstract. The classical Gomory-Hu algorithm aims for finding, for given input flows, a network topology for data transmission and bandwidth of its channels which are optimized subject to minimal bandwidth criteria. In practice, it may occur that some channels between nodes of the network are not active. Ignoring such channels using the topology obtained be the Gomory-Hu algorithm will not lead to an optimal flow-rate.

In this paper the focus is on a modified algorithm taking into account deficient channels. While the classical algorithm generates a sequence of ring subnets, in our modified version the use of deficient channels is checked at intermediate stages in each cycle of the algorithm. When forming ring subnets, the availability of new channels to be introduced into the ring subnet is checked and in the case of unavailability another ring closest to the optimal one is selected. The network optimized by this modified algorithm guarantees the transmission of the maximum input stream.

Keywords: Network topology · Channel capacity · Gomory-Hu algorithm

1 Preamble

From a historical point of view, networking, data transmission and distributed processing developed as a result of scientific and technological progress. Modern networks connect a huge amount of computers and other devices via communication channels. In view of their growing complexity the problem of optimizing their performance has become more and more important.

P. Gaj et al. (Eds.): CN 2020, CCIS 1231, pp. 3–13, 2020.
https://doi.org/10.1007/978-3-030-50719-0_1

In mathematical terms, a network corresponds to a (weighted) graph, directional or non-directional. Network servers, routers, etc. are vertices, and communication lines (channels) are the edges of such a graph.

2 Introduction: Problems of Network Performance Optimization

A computer network is a complex combination of data terminal equipments, different data communication equipments such as routers, physical environment, channels, application processes, data flows, communication and routing protocols, etc. Therefore, optimization of network performance is a complicated task requiring from developers to assess the expected performance.

For studying such problems it is advisable not to create a real physical network[1] but to use a mathematical model, on the basis of which it will be possible to judge the efficiency of future networks and to decide whether to realize them or how to improve their topology.

A mathematical description of a network is typically based on a graph-theoretical model, where the set of vertices represents the nodes of the network, and the set of edges represents the channels connecting them. Network topologies are traditionally described as undirected graphs without loops or multiple edges. In Fig. 1 we visualize typical topologies.

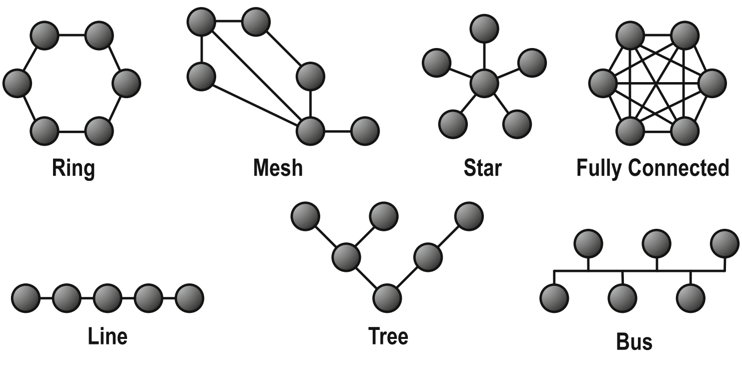

Fig. 1. Typical network topologies

[1] Even for smaller networks, physical modeling requires large effort, time and considerable material costs. Thus, the possibilities of physical modeling are rather limited. It only allows to investigate special settings where a small number of combinations of the relevant system parameters are taken into account.

The major criteria affecting efficiency are

– performance,
– reliability,
– security.

The performance of a network is affected by the number of users, transmission media, hardware and software limitations. It is measured in terms of

– transit time,
– response time,
– throughput,
– delay.

Finding an optimal topology for a designed network taking the above criteria into account, requires a multi-criteria analysis which is not easy to formalize. This is due to the fact that these criteria have different (often contradictory) effects on the analyzed object. For example, increasing reliability leads to redundant components, connections require better equipment, etc.

Many works were devoted to the design of different types of network topologies, examples of which can e.g. be found in [4,6], where methods and algorithms for optimization are systematized and analyzed. The principles underlying the Gomory-Hu topological design are described in [2,3,6]. Gomory and Hu proposed an algorithm providing a synthesis of network topology and choice of channel capacities. The network designed according to this algorithm enables transmission of a maximum given input flow with a minimum required total capacity over the channels. In [7] a simulation model based on the Gomory-Hu algorithm (besides some other algorithms) was implemented. In [8], devoted to the embedding of virtual topologies in network clouds, the algorithmic steps proposed start with building the Gomory-Hu tree.

Optimization via the classical algorithm results in channel capacities which do not take into account particular transmission technologies in the different channels. But the capacity of the separate channels should be selected in accordance with the requirements of their transfer technology. An appropriately modified algorithm taking into account the requirements of the Dense Wavelength Division Multiplexing (DWDM) technology is presented in [1].

The aim of the present paper is to design a network topology with minimum excess capacity by modifying the Gomory-Hu algorithm for cases when a physical realization for some of the channels is not available or needs to be avoided.[2]

3 The Classical Gomory-Hu Algorithm Aiming for Optimizing the Network Topology and Selecting the Bandwidth of Its Channels

The input data is a set of nodes, together with the requirement of exchange of information and the intensity of flows that need to be provided between them. The algorithm assumes that this data are represented by a non-oriented weighted graph, The weights of the edges (=channels) correspond to the flows to be transmitted. The result of the optimization procedure is a weighted graph representing the topology of the network after optimization. The weights of channels represent the channel capacities.

Applying the classical Gomory-Hu algorithm one can find a network topology and the capacity of its channels for which transmission with maximum flow is ensured, and at the same time the weights of all edges (the required capacity of the communication channels) will be minimal. Let us discussed the application of the Gomory-Hu algorithm for network topology optimization and choosing capacity of its channels. The algorithm assumes that, according to certain rules, the input graph is divided into a set of graphs, of which represents some subnet. All subnets, with the exception of the last one, which can be a segment connecting two nodes, are ring subnets with the same weight value of each edge of the uniform ring. By superposition of all the resulting subnets one obtains an optimized network that will feature minimum total capacities of the channels (edges), while providing transmission of maximum flow. Found in the process of optimization the weights of the edges are equal to the required bandwidth of the channels. For better understanding the Gomory-Hu algorithm can be divided into two major stages:

(i) decomposition,
(ii) superposition.

[2] Nowadays there is a high risk of intentional damage caused by truncating communication lines. This problem is particularly relevant for backbone lines providers. For example, in 2017, there was the massive attack on Ukrtelecom's backbone lines serving eastern Ukraine. In two places a long-distance line was cut and cables were damaged [9].

According to the new General Data Protection Regulation (GDPR) [10] adopted in EU in 2018 it is prescribed to implement appropriate technical and organizational measures to ensure a level of security appropriate to the risk. In case of a physical or technical incident one should be able to recover all personal data in a timely manner. From our point of view, the best solution to provide these requirements when developing a backbone network topology is to exclude dangerous lines in the network, for example from the military area. Our modified algorithm will help to do this, although at the cost of slightly exceeding bandwidth.

(i) Decomposition can be described in following way:
  1. Specification of a weighted non-directed graph G_{in}, in which vertices of graph represent network nodes, and edges represent flows where the weights of the edges represent the required intensities of flows.
  2. Construction of the graph $A := G_{\text{in}}$.
  3. Decomposition of the graph A into
     - a ring graph SN_i which includes all vertices of the graph A, which have edges and assigning to each edge of the ring the weight $W_{\text{min}}/2$; where i is the cycle number and W_{min} is the minimal weight of the edges of the graph A;
     - a graph B which is obtained by subtracting the value W_{min} from the weight of each edge of graph A whose weight is positive.
  4. If the number of edges in the graph B is larger than one, we accept $A = B$ and go back to step 3; otherwise the decomposition is completed.

(ii) Superposition means constructing the output graph by integration of all graphs to which the input graph was decomposed in step 1. Let us illustrate the work of the algorithm on the example of a network topology optimization, network includes 7 nodes, among which should be transmitted streams described below in Example 1.

Example 1. The input data are the intensities of flows a_{ij} between the i-th and j-th nodes:

$$\begin{aligned} a_{12} &\Rightarrow 100\,\text{Tb/s}, \\ a_{14} &\Rightarrow 50\,\text{Tb/s}, \\ a_{15} &\Rightarrow 20\,\text{Tb/s}, \\ a_{23} &\Rightarrow 6\,\text{Tb/s}, \\ a_{25} &\Rightarrow 10\,\text{Tb/s}, \\ a_{46} &\Rightarrow 4\,\text{Tb/s}, \\ a_{67} &\Rightarrow 10\,\text{Tb/s}. \end{aligned}$$

Figure 2 shows the process and result of optimization based on the classical Gomory-Hu algorithm.

Let us discuss these results. The verification is performed for maximum flow, in our case it is $a_{12} = 100$ Tb/s. In the resulting network, this flow can be transmitted simultaneously in the following way:

$$\begin{aligned} a_{12} &\Rightarrow 75\,\text{Tb/s}, \\ a_{1765432} &\Rightarrow 3\,\text{Tb/s}, \\ a_{176542} &\Rightarrow 2\,\text{Tb/s}, \\ a_{1542} &\Rightarrow 5\,\text{Tb/s}, \\ a_{142} &\Rightarrow 15\,\text{Tb/s}. \end{aligned}$$

Here, $a_{1765432}$ denotes the path from node 1 to node 2 across nodes 7, 6, 5, 4 and 3, etc. The sum of these flows is 100 Tb/s. Thus, since the maximum input

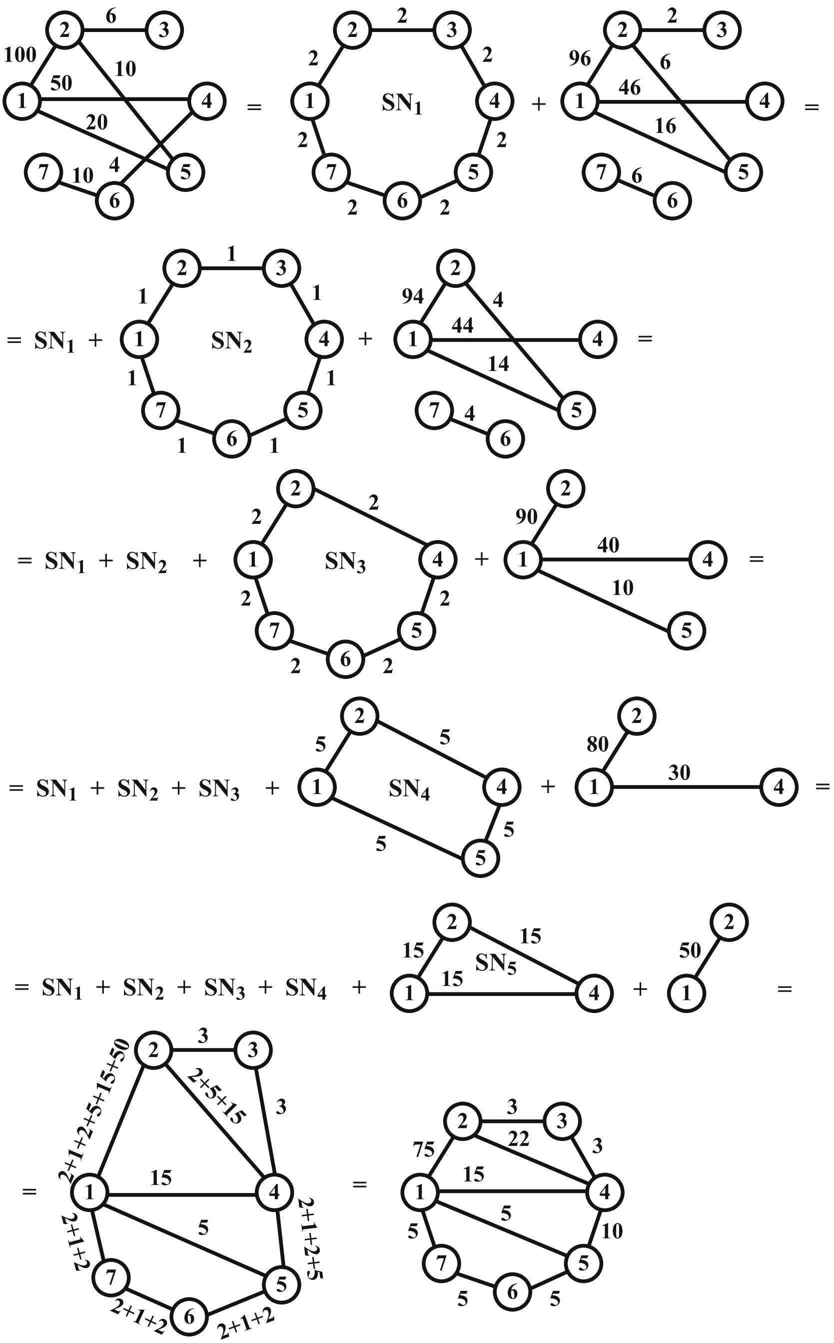

Fig. 2. (Example 1:) Results obtained by the classical Gomory-Hu algorithm.

stream is 100 Tb/s the network is able to transmit the maximum flow without any excess resources. i.e., all network resources are busy for transmission of maximum flow. The total capacity of all channels is 148 Tb/s.

Consider the case where for some reason it is not possible to provide some channel, or channels connecting the nodes of the outer ring. In this case it will not be possible to transmit the maximum input stream. The reason for this is that the algorithm provides an optimal solution which eliminates any redundancy. Therefore, rejecting any channel or reducing its bandwidth will not permit to achieve the desired goal.

4 A Modified Gomory-Hu Algorithm for Optimizing the Network Topology and Selecting the Bandwidth of Its Channels

Now we propose a modification of the classical algorithm in order to select the network topology and bandwidths of its channels according to the criteria of minimum bandwidth, in case of possible restrictions on the existence of certain channels. Additional input is a 'black list' of unavailable channels between the nodes of the outer ring. In the classical algorithm, ring subnets are iteratively constructed in course of the decomposition. In the modified algorithm, in each cycle it is checked whether a channel from the black list of unavailable channels is included in the ring subnet. If such a channel is not contained in the black list, the algorithm proceeds in the same way as the classical one.

Otherwise, we propose to use at this step of the algorithm the topology of the previous ring. Obviously, there will be some redundancy, but it will offer a solution that can be implemented. The modified algorithm can be described as follows.

1. Specification of a weighted non-directed graph G_{in}, in which vertices of graph is network nodes and edges are flows, weights of the edges represent the required intensities of flows.
2. Construction of graph $A := G_{\text{in}}$.
3. Find the minimal weight of the edges of the graph $A - W_{\text{min}}$.
4. Decomposition of graph A into:
   - a ring graph SN1 which includes all the nodes of graph A, and assigning to each edge of the ring the weight $W_{\text{min}}/2$;
   - a graph B, which is obtained by subtracting the value W_{min} from all edges of graph A whose weight is positive.
5. If the number of edges in graph B is equal one, then go to step 11, else we accept $A = B$.
6. Construction a ring graph SN_k which includes all the nodes of graph A, where k is the cycle number, which started from 2.
7. Comparing ring graphs $SN_k = SN_{k-1}$ to detect new edge. If the new edge does not appear go to 8. Otherwise check if the new edge is forbidden according to the list of forbidden channels. In this case, set $SN_k = SN_{k-1}$.

8. Find the minimal weight of the edges of the graph $A - W_{\min}$.
9. Decomposition of graph A into:
   - a ring graph SN_k and assigning to each edge of this ring the weight $W_{\min}/2$;
   - a graph B which is obtained by subtracting the value $W_{\min}$ from all edges of graph A whose weight is positive.
10. If the number of edges in graph B is larger than one, then we accept $A = B$ and go back to step 6.
11. Integration of all graphs of SN_k and graph B.

Example 2 (based on Example 1:). An illustration of the proposed approach is shown in the example of the implementation of the modified algorithm for the same input data as before for the classical version. Figure 3 shows the process and result of optimization based on our modified algorithm.

The verification is performed for maximum flow, in our case it is $a_{12} =$ 100 Tb/s. In the resulting network, this flow can be transmitted simultaneously in the following way:

$$\begin{aligned} a_{12} &\Rightarrow 75 \text{ Tb/s},\\ a_{1765432} &\Rightarrow \ 3 \text{ Tb/s},\\ a_{176542} &\Rightarrow \ 7 \text{ Tb/s},\\ a_{142} &\Rightarrow 15 \text{ Tb/s}. \end{aligned}$$

The sum of these flows is 100 Tb/s. Thus, this network also is able to transmit the maximum flow. The connection between nodes 1 and 5 is not involved, which cannot be physically implemented due to input conditions. The price for this is increasing the total capacity of all channels. In this example it equals 158 Tb/s. When optimizing according to the classical Gomory-Hu algorithm the total capacity of all channels was 148 Tb/s, but the classical algorithm does not take into account the constraints caused by unavailability of some channels.

For automated verification of results, the Gomory-Hu algorithm was used to determine the maximum flow in the network. The following are fragments from an implementation of the program for finding the maximum streams for the examples discussed above. The results for Examples 1 and 2 are shown in Fig. 4 and 5, respectively.

In the input data for the classical and modified algorithms, the maximum flow value to be transmitted between nodes 1 and 2 is 100 Tb/s. As can be seen from Figs. 4 and 5, the optimized topologies provide exactly this value.

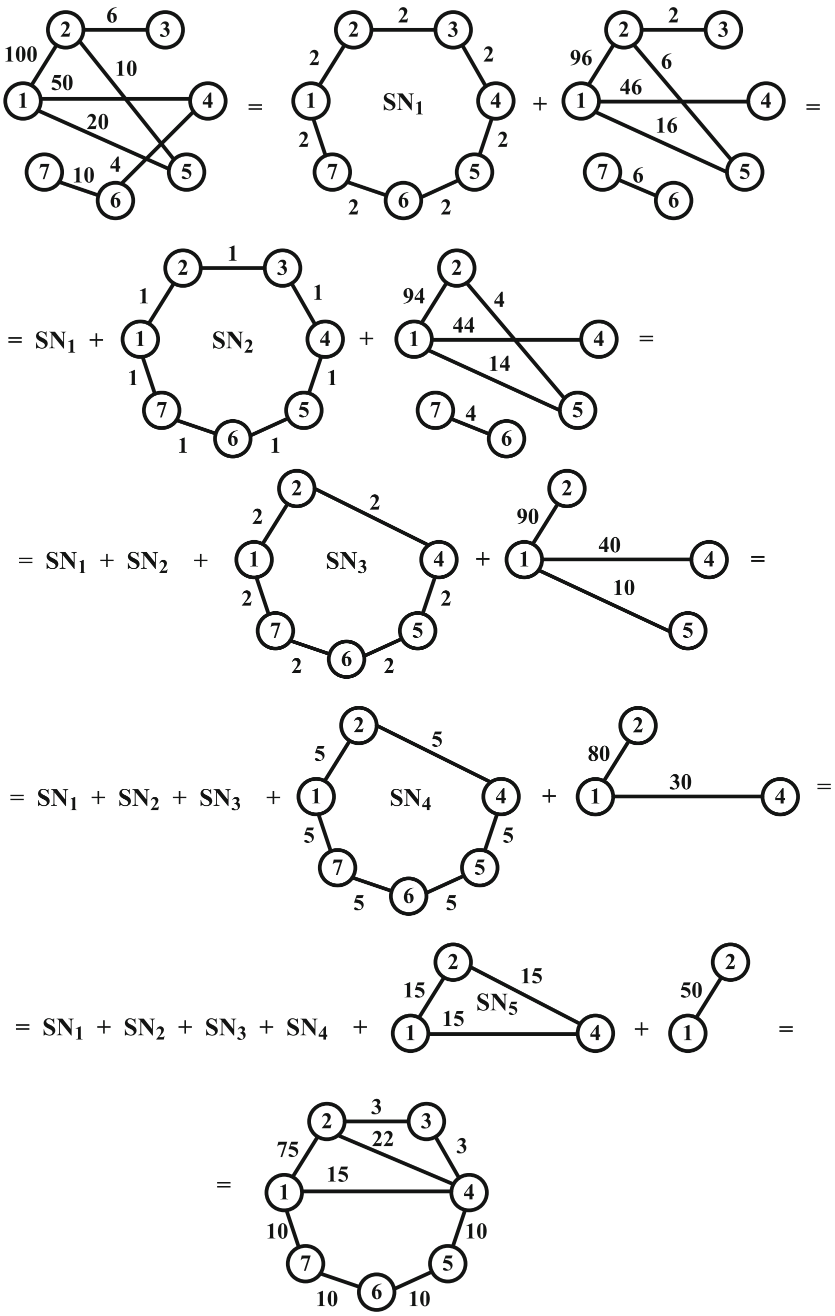

Fig. 3. (Example 2:) Results obtained by the modified Gomory-Hu algorithm.

```
Step 1: Vt = [ {{0},{1},{2},{3},{4},{5},{6}} ]
----------------------------------------------
Step 2: X = [ {0}, {1}, {2}, {3}, {4}, {5}, {6} ]
Step 3: G = [ {{0}}, {{1}}, {{2}}, {{3}}, {{4}}, {{5}}, {{6}} ]
          0  75  0  15   5  0  5
         75   0  3  22   0  0  0
          0   3  0   3   0  0  0
         15  22  3   0  10  0  0
          5   0  0  10   0  5  0
          0   0  0   0   5  0  5
          5   0  0   0   0  5  0
Step 4: s-t   = {{0}}-{{1}}
        max_f = 100
        A     = [ {0}, {2}, {3}, {4}, {5}, {6} ]
        B     = [ {1} ]
Step 5: Vt = [ {{0}}, {{2}}, {{3}}, {{4}}, {{5}}, {{6}},  {{1}} ]
               0  100
             100    0
```

Fig. 4. (Example 1:) Verification of the maximum flow for the classical algorithm.

```
Step 1: Vt = [ {{0},{1},{2},{3},{4},{5},{6}} ]
----------------------------------------------
Step 2: X = [ {0}, {1}, {2}, {3}, {4}, {5}, {6} ]
Step 3: G = [ {{0}}, {{1}}, {{2}}, {{3}}, {{4}}, {{5}}, {{6}} ]
          0  75  0  15   0   0  10
         75   0  3  22   0   0   0
          0   3  0   3   0   0   0
         15  22  3   0  10   0   0
          5   0  0  10   0  10   0
          0   0  0   0  10   0  10
         10   0  0   0   0  10   0
Step 4: s-t   = {{0}}-{{1}}
        max_f = 100
        A     = [ {0}, {2}, {3}, {4}, {5}, {6} ]
        B     = [ {1} ]
Step 5: Vt = [ {{0}}, {{2}}, {{3}}, {{4}}, {{5}}, {{6}},  {{1}} ]
               0  100
             100    0
```

Fig. 5. (Example 2:) Verification of the maximum flow for the modified algorithm.

5 Conclusion and Outlook

The proposed modified Gomory-Hu algorithm offers a solution for finding the network topology and bandwidth of its channels, when for some reasons certain channels are not available. A network optimized in this way still guarantees the transmission of the maximum input stream.

Further work on this topic (see [5] and Acknowledgement below) will include a study of complexity issues, also based on a systematic implementation, preferably using the graph-theoretical features available in the computer algebra system Maple.

Acknowledgement. (i) This work was realized within the framework of the Jean Monnet 2019 program under Erasmus+'611692-EPP-1-2019-1-UAEPPJMO-MODULE Data Protection in EU'.

It is also to be considered as a first step in a cooperation project under preparation, following up the joint Ukraine-Austria R & D project 'Traffic and telecommunication networks modelling', project No. UA 10/2017/0118U001750, M-130/2018 (cf [1]).

(ii) The authors thank Nadyja Popovuch, student at Lviv Polytechnic National University, for providing figures.

References

1. Auzinger, W., Obelovska, K., Stolyarchuk, R.: A modified Gomory-Hu algorithm with DWDM-oriented technology. In: Lirkov, I., Margenov, S. (eds.) LSSC 2019. LNCS, vol. 11958, pp. 547–554. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-41032-2_63
2. Ford, L.R., Fulkerson, D.R.: Flows in Networks. Princeton University Press, Princeton (1962)
3. Gomory, R.E., Hu, T.C.: Multi-terminal network flows. J. Soc. Ind. Appl. Math. **9**(4), 551–570 (1961)
4. Kasprzak, A.: Exact and approximate algorithms for topological design of Wide Area Networks with non-simultaneous single commodity flows. In: Sloot, P.M.A., Abramson, D., Bogdanov, A.V., Gorbachev, Y.E., Dongarra, J.J., Zomaya, A.Y. (eds.) ICCS 2003. LNCS, vol. 2660, pp. 799–808. Springer, Heidelberg (2003). https://doi.org/10.1007/3-540-44864-0_82
5. King, V.: Fully dynamic connectivity. In: Kao, M.Y. (ed.) Encyclopedia of Algorithms. Springer, Boston (2015)
6. Larsson, C.: Design of Modern Communication Networks. Methods and Applications. Elsevier Ltd., Amsterdam (2014)
7. Tapolcai, J.: Routing algorithms in survivable telecommunication networks. Ph.D., Thesis Summary, Budapest, Hungary (2004)
8. Xin, Y., Baldine, I., Mandal, A., Heermann, C., Chase, J., Yumerefendi, A.: Embedding virtual topologies in networked clouds. In: CFI 2011 Proceedings of the 6th International Conference on Future Internet Technologies, Seoul, Korea, pp. 26–29. ACM New York (2011)
9. https://www.rbc.ua/ukr/news/ukrtelekome-zayavili-massovoy-atake-magistralnye-1494495888.html
10. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, repealing Directive 95/46/EC (General Data Protection Regulation). https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679