

Multi-Hour Design of Dynamically Reconfigurable MPLS Networks^{*}

Balázs Gábor Józsa^{1,2}, Dániel Orincsay^{1,2}, and Levente Tamási²

¹ Budapest University of Technology and Economics, Department of Telecommunication and Media Informatics, High Speed Networks Laboratory, H-1117 Magyar tudósok körútja 2, Budapest, Hungary
{jozsa, orincsay, tamasi}@tmit.bme.hu

² Ericsson Research Hungary, Traffic Analysis and Network Performance Laboratory, H-1300 POB 107, Budapest, Hungary
{Balazs.Jozsa, Daniel.Orincsay}@ericsson.com

Abstract. This paper proposes an algorithm for cost-efficient network design. The two main network components are the routers and the links connecting them. As these network elements have discrete capacity values, their costs are modeled by stepwise functions. In the current interpretation of the network design problem the subproblems of topological design, resource dimensioning, and routing of traffic flows are handled simultaneously. Using new network technologies (e.g., MPLS) the operators have the opportunity to reconfigure their networks dynamically. By using this function the periodic (e.g., daily, weekly) changes of traffic volumes and directions can be taken into consideration during network design resulting in lower deployment costs. The algorithm proposed for multi-hour design is based on an efficient, published method. The efficiency of the new algorithm is demonstrated by simulation on random and real network topologies with the help of two reference algorithms.

Key words. MPLS, traffic engineering, network planning, multi-hour design.

1 Introduction

Nowadays, when the number of users of networking applications increases enormously, the management of network resources is a hot-topic issue. On the other hand, as technology develops, the capacities of network devices become larger and larger; a few years ago 155 Mbps links were general but now 10 Gbps links are also available. However, the need for bandwidth increases more rapidly than capacities of routers and links, since the bandwidth requirements of novel networking applications are higher and higher. Further, sometimes it is not worth extending the existing networks since the old technology that does not support novel applications or the extension is restricted by physical limitations. For these reasons the establishment of new networks using new technologies is essential. Thus, Internet service providers (ISPs) can make use of an algorithm that can solve the problem of network design aiming at reducing the deployment cost.

^{*} This work was supported by the Ministry of Education, Hungary, under the reference No. IKTA-0092/2002. <http://w3.ttt.bme.hu/ikta-2002>

An important issue considering the profitability of ISPs is how economically they can utilize their resources. In order to reach the optimal network performance, administrators should have full control over traffic flows. Fortunately, using novel network technologies, e.g., multiprotocol label switching (MPLS) [11], several traffic engineering [1] tools are available for the operator. For example, the paths of traffic flows can be given explicitly and can be reconfigured without the interruption of traffic [6]. The method of reconfiguration according to the daily and/or weekly traffic changes is called *capacity management* in a recommendation by the Telecommunication Standardization Sector of International Telecommunication Union (ITU-T) [4]. A possible approach of capacity management is multi-hour design (MHD). Multi-hour design—as opposed to single-hour design (SHD), which results in a network that is dimensioned for maximal (busy hour) traffic demands—takes the periodic change of traffic volumes and directions into account by partitioning the whole time scale into several intervals and calculating the maximal traffic demands separately for each interval. As the maximal demand between different node pairs may occur in different intervals, the capacities of network devices may be smaller than in the case of single-hour design, resulting in lower deployment cost. This design approach takes the ability of reconfiguration into account already at the network design phase.

In this study the relation between capacity and cost values tries to represent well the real situations, where network devices are built up from smaller modules with discrete capacity values. Thus, stepwise cost functions are used, which makes the complexity of the design task higher implying the need for a heuristic approach. This paper proposes an algorithm for multi-hour design problem which is based on the algorithm for single-hour design presented in [7]. Although MPLS is assumed as underlying network technology throughout this paper, the algorithm to be presented can be used for any technology supporting the traffic engineering features described above.

Similar problem for ATM networks was already discussed in the literature. In [2] the problem is divided into three sub-problems and during the iterative solution two of them are considered as static at one time. Medhi has several methods [8–10] for the problem, moreover he also investigated the issue of protection. The novelty compared to the previous approaches is that our algorithm uses stepwise cost functions and it does not restrict the paths between node pairs to pre-determined path sets.

The rest of the paper is organized as follows. The next section describes the applied models and gives the problem definition. In Section 3 the proposed algorithm is detailed. Then the simulation environment is presented and the numerical results are evaluated in Section 4. Finally, Section 5 contains the concluding remarks.

2 Problem Statement

This section presents the used interpretation of cost-optimal multi-hour network design problem. First, the network, traffic, and cost models are described. Then the problem formulation is given, finally the issue of complexity is discussed.

2.1 Network Model

The network is modeled by a directed graph where the routers and the links between them are represented by a set of nodes and a set of edges, respectively. Duplex links are supposed that are represented by oppositely directed edge pairs, where the two edges corresponding to a given link must have the same capacity values. For the easier handling of capacities of routers, the nodes are substituted by virtual links (see Fig. 1). A particular node n is decomposed into two nodes n_{in} and n_{out} belonging to the incoming and outgoing traffic of node n , respectively. Thus, n_{in} is connected only to the incoming links, while n_{out} is connected to the outgoing links, and node n is represented by the virtual link between n_{in} and n_{out} that the transit traffic must traverse. In this way the capacity constraint of nodes can be taken into consideration easily.

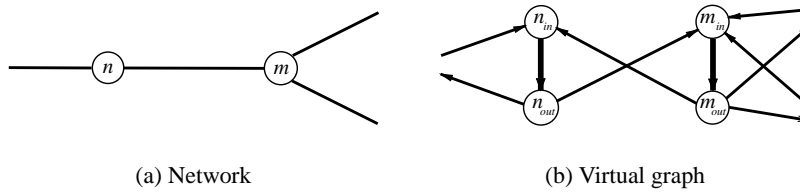


Fig. 1. Mapping the network onto a directed graph using virtual links.

2.2 Traffic Model

In the case of multi-hour design the time-scale is divided into several intervals. In the time intervals different traffic volumes and distributions can be given by different sets of traffic demands. A particular traffic demand is described by its source and destination routers, and its required capacity. Generally, one unsplittable traffic demand is assumed per node-pair, however, the model can handle more parallel demands for the same node-pair (e.g., representing more traffic classes) as well.

Since capacitated traffic demands are assumed, mapping them onto the network means that the required amount of capacity has to be reserved on each link and router along their previously computed paths. In MPLS an explicitly determined label switched path (LSP) should be established for each traffic demand using, e.g., the traffic engineering extension of the resource reservation protocol (RSVP-TE). Obviously, the sum of capacity reservations on a device cannot exceed its maximum (reservable) capacity value.

2.3 Cost Model

Basically, cost is one of the most important factors during the process of network design, therefore the applied cost model has large influence on the accuracy and applicability

of the used approach. The cost/capacity dependencies of various network devices are generally described by cost functions. Various approaches co-exist in network design starting from the simplest linear cost model up to the complex *stepwise* cost model (see Fig. 2). When using linear functions the problem is simpler, however, this approximation of the cost/capacity relationship is very inaccurate. On the contrary, stepwise cost functions can model real situations well, however, the drawback of this approach is the much higher complexity. Since this study aims at developing a method that could also be applied in real-world design tasks, the more sophisticated model of stepwise cost functions is followed.

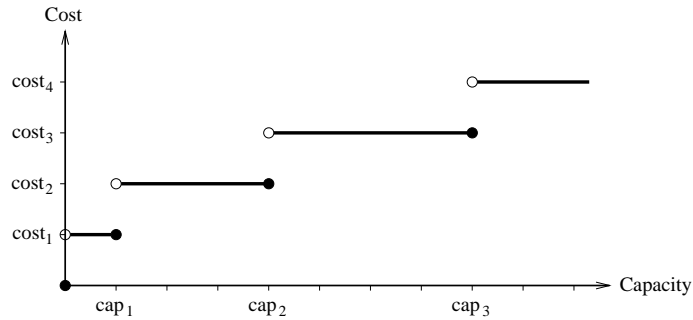


Fig. 2. Stepwise cost function example.

2.4 Problem Formulation

The formulation of the problem can be given by the sets of input and output data. The input sets are the following:

- locations of routers,
- possible links between the pairs of routers,
- individual cost functions for routers and links,
- traffic demand sets to be satisfied corresponding to different time intervals.

The following output data are determined as the result of the design process:

- capacity of each device,
- explicit path set for each traffic demand set.

2.5 Complexity

As it has been discussed in [7], the single-hour network design problem is already NP-hard [3] when using stepwise cost functions. Because of the more time intervals handled in multi-hour network design the problem is more difficult than in the case of single-hour network design. Since the latter one is a special case of the first one, the mathematical complexity of MHD is at least the same as that of SHD.

3 Multi-Hour Core Network Design Algorithm (CND^{MH})

In this section the new algorithm CND^{MH} proposed for cost-efficient multi-hour network design is detailed. First, the outline of the algorithm is described and then its three phases are presented.

3.1 Algorithm Outline

The multi-hour network design algorithm CND^{MH} has the same three phases as its single-hour design version CND presented in [7]. Naturally, the particular phases are extended due to the more sets of traffic demands corresponding to the different time intervals. The first phase is the initial capacity estimation (ICE^{MH}) that tries to foresee the approximate capacity needs. Then the iterative routing optimization (IRO^{MH}) follows, whose task is to provide a feasible network configuration using the output of ICE^{MH}. The last phase called posterior capacity refinement (PCR^{MH}) serves for finalizing the actual network configuration to achieve a cheaper solution. Note that IRO^{MH} is the main phase that can give a valid solution for the problem by itself, while the use of ICE^{MH} and PCR^{MH} is optional. However, it is worth using them as they contribute significantly to the quality of the solution. It is also important that in the special case when the number of time intervals equals to 1 all phases of CND^{MH} operate in the same way as their single-hour versions.

3.2 Initial Capacity Estimation (ICE^{MH})

The goal of ICE^{MH} is to foresee the necessary capacities of network devices by analyzing the sets of traffic demands. This phase provides a partially dimensioned network for IRO^{MH}, therefore the remaining tasks are to finish dimensioning of the devices and to route the sets of traffic demands corresponding to the different time intervals. Although ICE^{MH} does not give a feasible solution, it specifies a good starting state based on a global view of the problem instance.

The operation of ICE^{MH} is based on the routing of traffic demands one by one in random order. All traffic demand sets are routed in a predefined number of rounds using shortest path routing. We suppose that if all the random accommodations needed a certain amount of capacity on a particular device, then it is probable that also the optimal accommodation needs so much capacity. Therefore, the capacity values of the devices are specified in such way that they get the lowest value of the arising ones in the above accommodations.

3.3 Iterative Routing Optimization (IRO^{MH})

IRO^{MH} is the main phase of the algorithm, i.e., it can provide a full solution for the problem without the use of ICE^{MH} and/or PCR^{MH} phases. The base of this phase is an algorithm that is capable of routing a given set of traffic demands in a capacitated graph. In this paper—as in [7]—the algorithm proposed in [5] is used, which performs well in

terms of *feasibility*, namely, it is very probable that it finds a solution for a given problem instance provided that a solution exists. However, this applied routing optimization algorithm can be substituted by any other one solving the same task.

IRO^{MH} is based on an iteration having two steps. In the first step the traffic demand sets are tried to be routed in the graph while taking the actual capacity constraints into account. If the algorithm terminates with success for all traffic demand sets IRO^{MH} finishes. Otherwise, in the second step the capacity of a particular device is increased by one capacity step, and the first step follows again. The selection of the device to be enlarged is done by routing the rest of the traffic demands without capacity constraints for each time interval and then the device having the maximal capacity excess—that is averaged for the different time intervals—is chosen.

3.4 Posterior Capacity Refinement (PCR^{MH})

Although a feasible solution is available after the IRO^{MH} phase, this result can be improved with the help of PCR^{MH}. This phase is based on a local search procedure, i.e., the process concentrates only on one part of the network at one time. The idea behind PCR^{MH} is to reduce the capacity of such devices whose *relative step utilization* is low. The relative step utilization refers to the ratio of necessary and total capacity values on the actual capacity step (corresponding to the actual device cost).

PCR^{MH} starts with the sorting of the devices by their relative step utilizations—that are averaged for the different time intervals—and then the first device is chosen. The capacity of the actual device is decreased by one step and the sets of traffic demands are tried to be accommodated by IRO^{MH} under these tighter capacity conditions. If all sets can be routed then PCR^{MH} restarts, otherwise the capacity of the current device is reset and the next device is tried to be shrunk. After the last device has been tried to be shrunk without success PCR^{MH} finishes.

4 Results

This section contains the performance evaluation of the proposed algorithm CND^{MH} which was carried out by means of simulation. First, the process of problem instance creation is described including the generation of network topologies, traffic demand sets, and cost functions. Then the reference algorithms used for the comparison are introduced. Finally, the numerical results and their analysis are presented.

4.1 Problem Instance Generation

To ensure the reliability of simulations many different problem instances were investigated. The automated process of problem instance generation was similar to the one applied in [7]. However, in case of traffic demand sets a more complex generation method had to be used since more time intervals are distinguished. Note that CND^{MH} is able to handle any kind of problem instance that can be described by the model introduced in Section 2.

Network Topology. Generally, the task of network design is to plan possible future networks, thus the main focus of examinations was on random topologies. Besides, we also considered two existing real-world topologies (see Fig. 3). To create random topologies the random graph generation method published in [5] was used that tries to model the characteristics of real networks. In this study we concentrated on 25-node networks, however, we present results for 15-node and 35-node network sizes as well. The initial average nodal degree of topologies was set to 5.

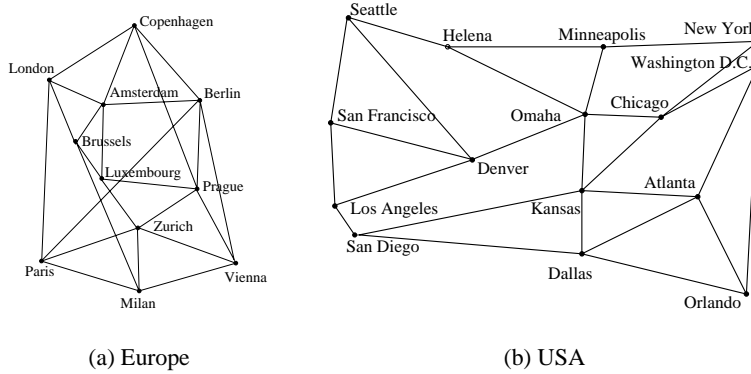


Fig. 3. The investigated real network topologies.

Traffic Demand Sets. The main characteristic of the multi-hour design problem is that the bandwidth demand of traffic flows between node pairs is varying in time. In order to create more traffic demand sets we divided the initial network topology into certain number of regions in the following way. First, the center of the topology was determined as a circle shape area. Its radius was set based on the expected number of nodes per region value that is the total number of nodes divided by the number of regions. Then the remaining topology was divided into angle ranges around the center, and the out-of-center nodes within the same angle range were assigned to one certain region. In a particular time interval each region was matched with exactly one other region. One part of the sum outgoing traffic of a given region specified by parameter Δ flowed towards its dedicated region pair, while the remaining part of traffic was distributed evenly among all regions (including itself as well as its region pair). In this way, $\Delta = 0$ means that there was no extra traffic between the specified region pairs and $\Delta = 100\%$ refers to the case when the total traffic flowed between the specified region pairs. When calculating the traffic demands for a given time interval the original traffic generation method detailed in [7] was used as first step, then the values for a given region were scaled based on parameter Δ so that the sum of traffic did not change. The difference between time intervals was based on the varying matching of regions. In this paper both the number of time intervals and the number of regions were set to 6. Since we had no information available about realistic traffic distributions—moreover traffic modeling is a huge research area that is out of the scope of this paper—different possible cases were examined with the help of parameter Δ that was shifted from 0% up to 100% by steps of 25%. The traffic volume was varied within a wide interval with the help of an input parameter representing the required average traffic per link value

(referred as ‘traffic volume’ in tables). Further, the traffic volume was constant in the different time intervals of a particular problem instance.

Cost Functions. During the investigation of CND^{MH} the same cost model was followed as in case of the original CND , in this way the same cost function generation was used (see [7]). In case of routers three capacity levels were distinguished specifying the maximum capacity to be handled (see Table 1). The cost functions of links were based on the standard capacity values of STM and it was supposed that the deployment of two parallel devices having the same capacity costs less than the device of the next capacity level (see Table 2). Note that the capacity values were modified randomly for each network device so that they would have unique cost functions.

Table 1. Costs of the routers.

capacity	cost (unit)
1 Gbps	5
10 Gbps	15
100 Gbps	60

Table 2. Costs of the links.

capacity	cost (unit)
155 Mbps	3
310 Mbps	6
622 Mbps	9
1244 Mbps	18
2.5 Gbps	27
5 Gbps	54
10 Gbps	81
20 Gbps	162

4.2 Reference Algorithms

Since information on the absolute optimal solutions of the problem instances were not available, two reference algorithms were used for evaluating the proposed algorithm CND^{MH} . Both the two references are simple extensions of CND . In this way the cost savings using CND^{MH} for multi-hour network design compared to CND can be investigated. After the reference algorithms determine the capacities of devices, the routes are calculated for each time interval with the help of global path optimizer [5] used in IRO^{MH} phase of CND^{MH} .

Traffic Bandwidth Maximizer (TBM). The main idea behind TBM is that the maximal traffic bandwidth requirement arising in the different traffic demand sets is considered between each node pair. Then the original CND is executed for the maximized traffic demand set. As the maximized traffic demands have at least the same bandwidth requirements as the individual traffic demands corresponding to the different time intervals, the designed network can surely accommodate every demand set. An important advantage of TBM—resulting from the bandwidth maximization—is that a feasible solution based on static routes can be provided for the case when path reconfiguration cannot be performed.

Link Capacity Maximizer (LCM). LCM handles the different time intervals separately, which means that it designs one particular network—with the help of CND—for each time interval independently from the other ones. After that the maximal arising capacity values are taken for every network device. In this way, the resulting network configuration can fulfill the capacity requirements of all traffic demand sets.

4.3 Numerical Results

This section presents the numerical results of the performed simulations. 5 different network topologies and 3 different traffic situations per topology were investigated for each problem class specified by network size, traffic volume, and traffic distribution value Δ . Due to the stepwise nature of the cost functions the size of the confidence interval of the results were relatively large, however, the tendencies were the same for each particular problem instance. Since the problem to be solved is cost optimization, the main focus of the investigations was on the total network cost. On the other hand, the running time was also examined to get a clear picture about the applicability of the algorithm in real situations.

The applied cost functions were based on cost units as it was discussed in Section 4.1, because we concentrate on the ratios of the cost values of different devices instead of the real commercial prices. When comparing the total cost values of the different design algorithms, the ratios were considered in percentage where 100% meant the best solution for the given problem class.

Table 3 shows the total cost values for 25-node networks. The results correspond to 10 different traffic volumes and 5 different Δ values (referring to the different traffic distributions). As one can see CND^{MH} gave the best solution almost in all cases. Exceptional cases can be found at $\Delta = 0$ setting, when TBM was slightly better than CND^{MH}. The largest differences between CND^{MH} and the reference algorithms can be observed at $\Delta = 100\%$: TBM and LCM were worse than CND^{MH} by 44% and 18% on average, respectively. Although we have no information about real traffic models, we suppose that real situations correspond to $25\% \leq \Delta \leq 75\%$ values, where the use of CND^{MH} results in a performance gain of 4–39% compared to the references. The ratios of solutions of different algorithms varied a bit as the traffic volume changed between 200 and 5500 Mbps, however, the tendency was clear.

In Table 4 the results (averaged for all traffic volumes) of different network sizes as well as real network topologies can be seen. The first observation is that CND^{MH} was the best for 15-node networks also at $\Delta = 0$. Another thing to note is that the performance gain attained by using CND^{MH} was slightly reduced when increasing the network size in general. In case of real networks—according to the results of the similar size random networks—CND^{MH} overperformed the references also in the case of evenly distributed traffic ($\Delta = 0$). The improvement gained by applying CND^{MH} varied between 7% and 35% compared to the references.

Considering the running time values—that were measured on a Sun Ultra Enterprise 420R with an Ultra II 450Mhz processor and 1GByte memory—it can be seen that for 15-node networks (as well as for the real networks) the time consumption is very low. Further, the worst running time value for 35-node networks is only about 1 hour, which is still moderate for an off-line network design tool. One can note that the running time

Table 3. Ratio of total network costs for 25-node networks (%).

traffic volume (Mbps)	algorithm	Δ				
		0%	25%	50%	75%	100%
200	CND ^{MH}	100.10	100.00	100.00	100.00	100.00
	TBM	100.00	104.67	120.70	133.22	137.66
	LCM	105.62	105.31	107.85	110.48	115.09
300	CND ^{MH}	101.35	100.00	100.00	100.00	100.00
	TBM	100.00	104.40	116.53	137.65	142.89
	LCM	105.75	107.68	106.69	117.31	119.47
500	CND ^{MH}	101.16	100.00	100.00	100.00	100.00
	TBM	100.00	105.58	118.99	140.00	150.88
	LCM	109.28	109.18	109.80	117.80	121.07
800	CND ^{MH}	100.00	100.00	100.00	100.00	100.00
	TBM	100.62	106.04	125.08	142.19	142.44
	LCM	108.66	108.70	108.80	113.03	117.25
1200	CND ^{MH}	101.20	100.00	100.00	100.00	100.00
	TBM	100.00	107.33	121.61	133.52	138.99
	LCM	105.24	104.79	107.34	113.59	122.16
1700	CND ^{MH}	101.57	100.00	100.00	100.00	100.00
	TBM	100.00	101.93	116.76	140.01	140.23
	LCM	106.86	104.20	108.36	115.67	120.27
2400	CND ^{MH}	102.32	100.00	100.00	100.00	100.00
	TBM	100.00	104.36	119.32	138.53	150.51
	LCM	109.67	107.40	107.14	117.42	118.37
3200	CND ^{MH}	100.00	100.00	100.00	100.00	100.00
	TBM	101.85	102.37	121.43	150.75	145.10
	LCM	106.98	108.57	108.69	115.31	120.04
4200	CND ^{MH}	100.01	100.00	100.00	100.00	100.00
	TBM	100.00	104.10	116.24	143.43	146.03
	LCM	107.40	106.33	110.72	113.95	113.71
5500	CND ^{MH}	101.45	100.00	100.00	100.00	100.00
	TBM	100.00	100.37	114.58	131.66	147.31
	LCM	102.89	107.25	110.70	114.14	114.60
Average	CND ^{MH}	100.67	100.00	100.00	100.00	100.00
	TBM	100.00	104.11	119.12	139.09	144.20
	LCM	106.57	106.94	108.61	114.87	118.20

Table 4. Cost and running time results for different network sizes.

network size	algorithm	Δ					time (minute) min–max (avg)
		0%	25%	50%	75%	100%	
15 nodes	CND ^{MH}	100.00	100.00	100.00	100.00	100.00	0–3 (< 1)
	TBM	100.17	107.32	123.10	145.80	152.23	0–1 (< 1)
	LCM	105.75	108.59	113.76	125.29	126.65	0–2 (< 1)
25 nodes	CND ^{MH}	100.67	100.00	100.00	100.00	100.00	1–20 (7)
	TBM	100.00	104.11	119.12	139.09	144.20	1– 8 (2)
	LCM	106.57	106.94	108.61	114.87	118.20	1–37 (6)
35 nodes	CND ^{MH}	104.83	100.00	100.00	100.00	100.00	1–61 (25)
	TBM	100.00	102.62	116.83	134.33	149.62	1–29 (10)
	LCM	108.25	106.80	107.79	112.34	121.88	1–58 (30)
Europe 11 nodes	CND ^{MH}	100.00	100.00	100.00	100.00	100.00	0–1 (< 1)
	TBM	111.42	106.86	120.06	129.85	128.74	0–1 (< 1)
	LCM	126.74	122.54	127.83	128.40	115.19	0–1 (< 1)
USA 15 nodes	CND ^{MH}	100.00	100.00	100.00	100.00	100.00	0–2 (< 1)
	TBM	108.08	109.14	115.12	126.67	134.96	0–1 (< 1)
	LCM	121.11	121.98	123.12	123.69	117.57	0–2 (< 1)

of TBM is significantly lower than in case of the other two algorithms. The reason for this is that TBM performs the original CND only once, while LCM repeats CND for all time intervals and CND^{MH} has more sub-processes that are multiplied due to the more sets of traffic demands.

5 Concluding Remarks

This paper addressed the problem of multi-hour design of MPLS networks. In the current interpretation of the problem the time-scale is divided into several intervals, and different sets of traffic demands are given describing the traffic volumes and distributions for all time intervals. The cost/capacity relations of network elements such as routers and links are modeled by stepwise functions. The advantage of this approach is that it can approximate real situations well. On the contrary, its drawback is the high mathematical complexity, which makes it reasonable to handle the problem with the help of a heuristic approach. We proposed an algorithm for the multi-hour network design problem combining the benefits of global and local search methods.

The performance of the novel algorithm was investigated with the help of simulations. Various size random networks as well as real world topologies were considered. In order to get a reliable picture, various traffic situations were examined starting from the evenly distributed one up to the fully polarized one. Further, two simple algorithms—based on simple extensions of the original single-hour design algorithm—were introduced as reference. Generally, the new proposal overperformed the reference algorithms. In exceptional situations, in the case of evenly distributed traffic combined with larger networks the performance of the new algorithm sometimes fell behind, compared to the first reference algorithm. However, this does not contradict our expectations

since multi-hour design makes sense when significant difference can be observed between the traffic distributions of various time intervals. The running time of algorithms can be considered moderate also in the case of larger networks, regarding that they solve off-line network design task. In summary, we may say that the proposed multi-hour design algorithm can provide economical solutions with acceptable running times for the examined network situations.

Our future work can include the extension of the current algorithm so that it can handle backup paths as well. Further, in order to increase the reliability of simulation results more realistic traffic demand models should be used. Finally, a new reference algorithm optimized for multi-hour design should be investigated, which will possibly require adaptation in order to fit the problem definition applied (e.g., stepwise cost functions).

References

1. Awduche, D., Malcolm, J., Agobua, J., O'Dell, M., McManus, J.: Requirements for Traffic Engineering Over MPLS. Internet Engineering Task Force, Request for Comments: 2702 (1999)
2. Bauschert, T.: Multihour design of multi-hop virtual path based widearea ATM networks. Proc of the 15th International Teletraffic Congress (ITC-15), Washington DC, USA (1997) 1019–1029
3. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. Freeman, San Francisco, CA. (1979)
4. ITU-T: QoS Routing and Related Traffic Engineering Methods – Capacity Management Methods. Recommendation E.360.6, CA. (2002)
5. Józsa, B.G., Király, Z., Magyar, G., Szentesi, Á.: An Efficient Algorithm for Global Path Optimization in MPLS Networks. Optimization and Engineering, Vol. 2., No. 3. Kluwer Academic Publishers (2001) 321–347
6. Józsa, B.G., Makai, M.: On the Solution of Reroute Sequence Planning Problem in MPLS Networks. Computer Networks, Vol. 42., No. 2. Elsevier Science (2003) 199–210
7. Józsa, B.G., Orincsay, D., Kern, A.: On the Use of Routing Optimization for Virtual Private Network Design. In Proc. of the 7th IFIP Working Conference on Optical Network Design & Modelling (ONDM'2003), Budapest, Hungary (2003) 865–880
8. Medhi, D.: A Unified Approach to Network Survivability for Teletraffic Networks: Models, Algorithms and Analysis. IEEE Transactions on Communications, Vol. 42. (1994) 534–548
9. Medhi, D.: Multi-Hour, Multi-Traffic Class Network Design for VP-based Wide-Area Dynamically Reconfigurable ATM Networks. In Proc of IEEE INFOCOM'95, Boston, MA. (1995) 900–907
10. Medhi, D.: Some Approaches to Solving a Multi-Hour Broadband Network Capacity Design Problem with Single-Path Routing. Telecommunication Systems, Vol. 13. (2000) 269–291
11. Rosen, E., Viswanathan, A., Callon, R.: Multiprotocol Label Switching Architecture. Internet Engineering Task Force, Request for Comments: 3031 (2001)