University of Würzburg
Institute of Computer Science
Research Report Series

MPLS Traffic Engineering in OSPF Networks - a combined Approach

Stefan Köhler[1] and Andreas Binzenhöfer[2]

Report No. 304 February 2003

[1] Department of Distributed Systems,
Institute of Computer Science,
University of Würzburg, Germany
koehler@informatik.uni-wuerzburg.de
phone: +49 931 888-6651
fax: +49 931 888-6632

[2] Department of Distributed Systems,
Institute of Computer Science,
University of Würzburg, Germany

MPLS Traffic Engineering in OSPF Networks - a combined Approach

Stefan Köhler
Department of Distributed Systems,
Institute of Computer Science,
University of Würzburg, Germany
koehler@informatik.uni-wuerzburg.de
phone: +49 931 888-6651
fax: +49 931 888-6632

Andreas Binzenhöfer
Department of Distributed Systems,
Institute of Computer Science,
University of Würzburg, Germany

Abstract

One of the main reasons for the development of MPLS was the need for flexible Traffic Engineering (TE) in IP networks ([8], [1]). Additionally a lot of IP service providers have found that Traffic Engineering of their networks can have a tactical and strategic value [2]. The first part of the paper introduces a linear optimization approach for Traffic Engineering in MPLS networks. Most remarkable on the presented approach is the additive objective function, which optimizes both the maximum utilization as well as the average utilization. In practice, most of the existing OSPF or IS-IS networks will have a transition phase between the pure IGP routing and the MPLS driven approach. Thus, we extend our optimization approach from a regular IGP network to a mixed IGP-MPLS environment. To avoid configuration complexity and state space explosion in MPLS devices, it could also make sense to start with a MPLS environment and take advantage of the configuration simplicity and the state space reduction capability of IGP protocols. In addition the paper includes a performance comparison between default configured OSPF, weight optimized OSPF and pure MPLS networks and show the benefit of OSPF optimization and MPLS Traffic Engineering. Based on this comparison we can offer some valuable clues to decide in favor of a local or global TE system.

Keywords: MPLS, Routing, IGP, Optimization, Traffic Engineering, Linear Programming

1 Introduction

Developed in the late 90s by the Internet Engineering Task Force (IETF), Multi-protocol Label Switching (MPLS), is a network management protocol intended to integrate layer 2 information like bandwidth, latency or utilization into the IP layer. MPLS technology offers more flexibility by placing labels on IP packets and using label switched paths (LSPs) to transmit packets through the network. One of the most obvious advantages of MPLS is the possibility for traffic engineering in IP networks. Traffic Engineering is the process of controlling the way traffic flows through a network to optimize resource utilization and network performance. Traffic Engineering is needed mainly because current IGPs like

OSPF or IS-IS [12] are not traffic aware and always use the shortest paths to forward traffic.

Beside the simple configuration, this principle conserves network resources, but also causes the problems that the shortest paths from different sources overlap at some links, resulting in congestion on those links and that the traffic is not equally distributed in the network. This possibility leads to situations where the traffic from source to destination is blocked, while a longer path between these two routers is under-utilized.

To provide improved support for Traffic Engineering, the IETF introduced MPLS [8], Constraint-base Routing [10], enhanced link state IGPs [13] and the modification of the IGP metric [9]. MPLS avoids problems of IGP by extending the way to determine a path from source to destination with the following mechanisms:

1. specify the complete path
2. use the IGP protocol to determine the path between a given source and destination
3. or use Constraint-Base Routing.

This paper focuses on the first two MPLS mechanisms and the adaptation of the IGP metric and tries to compare and evaluate the ability of the two concepts for TE in IP networks. Besides the TE aspect of MPLS there certainly are several other interesting features like the support for QoS, that are worth of being discussed in detail, but are beyond the scope of this paper.

The paper is organized as follows: Section 2 introduces the linear optimization system for pure MPLS networks. In Section 3 the evaluation and results of the linear optimization approach are presented and compared to default and optimized IGP Routing. Section 4 combines the MPLS and the IGP Routing approaches to a common TE system and deals with the corresponding evaluation and results. Finally, Chapter 6 concludes the paper with a summary and an outlook.

2 Optimization problem formulation

Unlike IGP routing algorithms like OSPF or IS-IS the MPLS forwarding scheme offers greater flexibility to place traffic flows individually. Our goal is to minimize a combination of the maximum and average utilization by finding the optimal set of paths.

First we define our networks by three NxN matrices:

- a matrix c_{ij} describing the link capacity between routers i and j. Note that in asymmetric networks the following may be true: $c_{ij} \neq c_{ji}$
- a matrix f^{uv} containing the flow demands for every pair of nodes u and v. Each entry f^{uv} describes the end-to-end traffic demand from router u to router v.
- an optional matrix d_{ij} mirroring the physical delays or interface costs of link (i, j)

The corresponding objective function to minimize the maximum and the average utilization in a network is shaped as follows:

$$a_t t + \sum_{(i,j)} \sum_{u \to v} \frac{f^{uv} x_{ij}^{uv}}{c_{ij}} \ , \quad c_{ij} \neq 0 \tag{1}$$

While t represents the maximum link utilization the parameter a_t is called the weight factor. It is used to define the importance of the optimization of the maximum link utilization compared to the average utilization. The higher a_t is set the more important the maximum utilized link gets. The default value in this paper is set to $a_t = 1000$. The variable x_{ij}^{uv} is declared as a real number between 0 and 1. It defines the percentage of the flow $f^{uv} \geq 0$ routed through link (i, j).

A linear program (LP) on principle consists of two parts. An objective function and linear constraints consisting of equations and in-equations. In our example we want to minimize the objective function given in Equation 1. On that account one has to find typical features reflecting a real world network and translate them into mathematical linear constraints like the following:

- Capacity and Utilization constraints

  The main principle of capacity and utilization constraints is to define how much traffic one link may take. The capacity constraints therefore give a fixed hard limit:

  $$\sum_{uv} x_{ij}^{uv} f^{uv} \leq a_c c_{ij} \quad \forall (i, j) \tag{2}$$

  The parameter a_c thereby specifies the limit.

  The following utilization constraints are only used when $a_t \neq 0$. That is, when the maximum utilization is optimized. The constraint is based on the variable t:

$$\sum_{uv} x_{ij}^{uv} f^{uv} \leq \frac{tc_{ij}}{100} \quad \forall (i,j) \tag{3}$$

- Transport constraints

  The transport constraints ensure that a flow takes a well defined path(es) through the network. The transport constraints have to accomplish flow conservation:

$$\sum_{j=1,c_{ij}>0}^{N} x_{ij}^{uv} \leq 1 \tag{4}$$

  A flow from node u to node v has to originate from the source u to its next hop(s):

$$\sum_{i=1,c_{ui}>0}^{N} x_{ui}^{uv} - \sum_{i=1,c_{iu}>0}^{N} x_{iu}^{uv} \geq 1, \quad i \neq u \tag{5}$$

  Every router except the source and the destination has to pass on the packets of a specific flow uv:

$$\sum_{j=1,c_{ij}>0}^{N} x_{ij}^{uv} - \sum_{j=1,c_{ji}>0}^{N} x_{ji}^{uv} = 0 \quad \forall i \notin \{u,v\} \tag{6}$$

  Finally each flow must reach a destination router v:

$$\sum_{i=1,c_{iv}>0}^{N} x_{iv}^{uv} - \sum_{i=1,c_{vi}>0}^{N} x_{vi}^{uv} \geq 1, \quad i \neq v \tag{7}$$

- Delay and QoS constraints

  Besides the usual flow constraints modelling IP and MPLS networks we can introduce additional constraints to support Quality of Service (QoS) and thus keep the physical delay within a reasonable range:

$$\sum_{ij} x_{ij}^{uv} d_{ij} \leq a_r d_{min}^{uv} \tag{8}$$

  That is, the delay for any flow must not exceed a multiple a_r of the minimum d_{min}^{uv} of all possible physical delays d_{ij}.

In this paper we concentrate on pure MPLS systems without any QoS constraints. However, the approach can easily be extended by introducing overlay networks. These overlay networks split the network into classes. Each class leads to a new system of the three class depended NxN matrices. In a first approximation, these overlay networks can be solved independently and merged to the requested solution later.

There are no known supplementary constraints to avoid loops. However, the minimization of the average utilization in the objective function 1 already prevents cycles. The LP problem is finally given to and solved by a linear optimizer like CPLEX [11].

3 Performance evaluation of the MPLS Traffic Engineering system

First, we want to emphasize that there is an advantage in optimizing the maximum utilization and the average utilization at the same time. Other optimization approaches ([5], [14],) are using only one part of the objective function used in this paper. The chosen objective function fits the definition of Traffic Engineering given at the beginning of the paper very. The minimization of the maximum link utilization follows the performance principle and the reduction of the average utilization takes care of the network resources. Table 1 illustrates the differences between default OSPF and linear optimization focusing on average utilization, maximum utilization and finally on both at once.

Network	20				25			
	Def. OSPF	Avg. only	Max. only	Avg. & Max.	Def. OSPF	Avg. only	Max. only	Avg. & Max.
Maximum	0.992	0.992	0.436	0.436	0.807	0.783	0.372	0.372
Average	0.234	0.234	0.417	0.238	0.255	0.255	0.334	0.262

Table 1: Linear optimization in the 20 and 25 node network

The computation time for the different objective functions are almost equal. It takes less than a second on a Pentium III (500 MHz, 512 MB) PC to compute the results [1]. The results are based on the networks given in the Appendix. Using only a part of the proposed objective function did not lead to the desired results. The average utilization and the inherent shortest path principle of OSPF lead to similar results. The small but occurring differences are due to the the possibility of splits in MPLS networks. Reducing the maximum utilization only resulted in a very high average network utilization and did thus not sufficiently reduce the network congestion.

---

[1] The bottleneck for the pure MPLS approach is mainly the creation of the linear problem and not of the solution time.

Network	10	14	18	20	25
Flows	90	182	306	380	600
Maximum	7/0	5/0	3/0	53/12	43/3
Average	2/0	0/0	1/0	0/0	0/0
Both	6/0	6/0	2/0	16/1	17/0

Table 2: Number of total/multiple splits.

When applying linear optimization to our sample networks using the MPLS forwarding scheme we obtained a surprising result. Since all variables have been declared as real numbers one might expect a significant number of splits.

However, while the maximum utilization was reduced to the assumed optimum the splits failed to appear in the amount expected (see Table 2). Moreover, all except one split were simple splits. These phenomena are again based on the average optimization part of the objective function. The average utilization is optimal when all flows are using the shortest path. Thus splits are only needed to optimize the maximum utilization, otherwise they will be avoided. A fact which is particularly suitable for the upcoming mixed environment approach.

To evaluate the quality of our results we use the the following reference points

- Minimum: least utilized link
- Maximum: most utilized link
- Average: average utilization of the network

to compare our solution with a default configured and a weight optimized OSPF network. The standard configured OSPF network uses the default CISCO OSPF configuration [2] and the weight optimized OSPF network is determined by the OSPF weight optimization presented in [4].

Table 3 compares the unoptimized and optimized OSPF results to our new lower bound found by linear optimization of the MPLS networks.

---

[2] Cisco uses a reference bandwidth of 100 Mbps for cost calculation. The formula to calculate the cost is reference bandwidth divided by interface bandwidth. For example, in the case of a 10 Mbps Ethernet link, it's 100 Mbps/10 Mbps = 10.

Network	10			14			18		
	Def. OSPF	L.O. OSPF	MPLS	Def. OSPF	L.O. OSPF	MPLS	Def. OSPF	L.O. OSPF	MPLS
Minimum	0	0.65	0.603	0.011	0.011	0.011	0	0	0
Maximum	1.296	0.967	0.942	0.528	0.388	0.314	1.068	0.855	0.854
Average	0.817	0.829	0.827	0.189	0.186	0.193	0.166	0.179	0.176

Network	20			25		
	Def. OSPF	L.O. OSPF	MPLS	Def. OSPF	L.O. OSPF	MPLS
Minimum	0.007	0.007	0.007	0.001	0.001	0.001
Maximum	0.992	0.530	0.436	0.808	0.480	0.372
Average	0.234	0.233	0.238	0.255	0.256	0.262

Table 3: Reference points for the 10, 14, 18, 20 and 25 node networks

Note the largest improvement is between the OSPF and the optimized OSPF weight set. This indicates that optimized weights should be used in IGP networks instead of the default values. An even larger improvement is achieved by MPLS optimization. MPLS networks are consequently able to improve the network congestion beyond the possibilities of pure OSPF networks.

When implementing the presented approach in a real world environment the computation time is one of the important issues. Table 4 clearly emphasizes the advantage in computation time of the MPLS approach compared to the linear optimized OSPF approach. Therefore MPLS proves to be better suited for Traffic Engineering.

Network	10	14	18	20	25
L.O. OSPF	2.6	9.5	843	2762	14587
MPLS	< 1	< 1	< 1	< 1	< 1

Table 4: Computation time (in [s]) for linear optimized OSPF and MPLS

4 Synergy between MPLS and IGP networks

In this section we shift our focus away from the pure MPLS system. We now intend to integrate a technology like MPLS into an existing IGP network using overlay networks. MPLS is able to operate beside the IP protocol in the same network without any interference. One of the most obvious advantages by doing so is, that it greatly extends our possibilities for Traffic Engineering and offers new ways to reduce the number of MPLS VPNs. This reduction simplifies the configuration, avoids a possible state space explosion in MPLS devices and introduces a certain scalability for the intended network optimization.

4.1 MPLS Traffic Engineering in an OSPF environment

The idea behind all this is to remove the bottleneck traffic from the regular IP network and re-import it optimal distributed into a MPLS overlay network. While our primary objective still remains to reduce the overall network congestion we are furthermore looking for a way to be able to scale the achieved improvements. To do so, we will have to work out ways to combine MPLS and OSPF networks.
This section will therefore be concerned about how to:

1. Decompose a network
2. Design the MPLS overlay network
3. Re-assemble the network

To differentiate the following algorithm from the pure MPLS optimization approach we name the concept DDR (Decompose-Design-Re-assemble)-algorithm.

Decompose a network

We start of with an all OSPF or IS-IS network. It does not make much difference to our approaches whether we are dealing with an optimized or unoptimized set of weights. However, as we are aiming at preferably fast ways to optimize networks we will base our following results on unoptimized weight sets to avoid the time needed to optimize link weights as presented in [4] or [5]. At first, we have to choose a set of flows that will be taken out of the network. Therefore our algorithm uses the following two parameters:

- b: The number of bottlenecks regarded
- f_i: The number of flows considered

While we can simply take the b most utilized links as our bottlenecks we have to invest some more thoughts into searching the appropriate flows. Re-placing flows with a very great bandwidth proved to be too inflexible as that way the problem area was simply shifted to another location. Flows with a medium bandwidth still are not fine-grained enough to adequately solve the given problem. Finally, flows carrying a small bandwidth seem to be best suited to re-distribute traffic to other links. We therefore build a list of all occurring flows and sort it as follows:

- Primary sorting is done by bottlenecks used and independent of the flow bandwidth. That is we sort the list of flows descending by the number of bottlenecks they use.
- Secondary sorting is done by bandwidth. That is if two or more flows use the same number of bottlenecks we secondary sort this set of flows ascending by their bandwidth.

All we have to do now is to take the first f_i flows from this sorted list and use them to build up our MPLS overlay network. Note that if there are less than f_i flows using any of the b bottlenecks we simply take less flows into account.

Table 5 illustrates the list of the five best suited flows considering 20 bottlenecks in our 20 node sample network. The first three flows using two bottleneck are secondary sorted ascending by their bandwidth. The last two flows are the flows with the smallest bandwidth using exactly one of the b most utilized links again sorted by their bandwidth.

Flow	Path	Bottlenecks	Bandwidth
15-7	15;4;7	2	1.75
15-1	15;4;0;1	2	4.73
15-6	15;4;6	2	9.41
15-2	15;10;3;2	1	1.18
9-1	9;0;1	1	1.98

Table 5: The five best suited flows considering 20 bottlenecks in the 20 node network

Furthermore, we want to emphasize that the parameter f_i determines the number of flows we are regarding and does not represent the final number of flows f_o that will actually be re-distributed. This is due to the fact that depending on the specific network and the current OSPF cost matrix more or less flows will already take the optimal path through the network. If so, there is absolutely no sense in forcing the flow to take an alternative path or include it into our MPLS overlay network respectively.

Design the MPLS overlay network

The next step is building the MPLS overlay network. We use an illustrative example to describe the step in principle. In the following figures red is used to denote the unused capacities, blue stands for the bandwidth used by flows of the OSPF network and green denote the bandwidth used by flows that will be re-distributed into the MPLS overlay network.

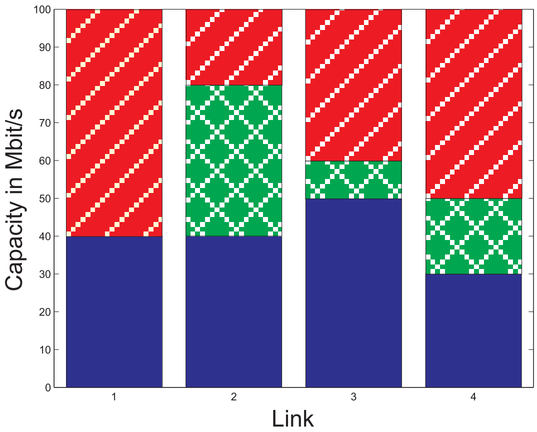


Figure 1: Utilizations in the initial OSPF network

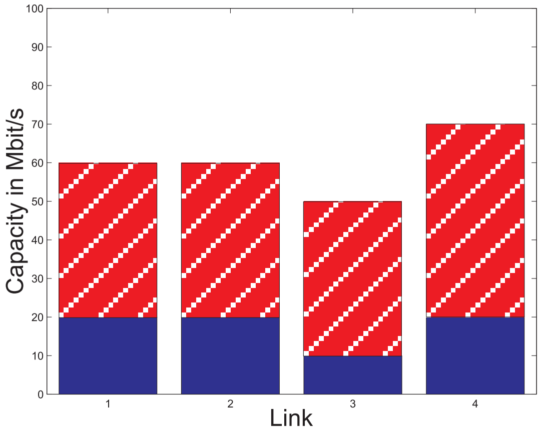


Figure 2: Optimized utilizations for the MPLS overlay network

We start off with an all OSPF network. At first, we have to choose a set of flows that will be taken out of the network and used in the MPLS overlay network later on. Figure 1 displays four exemplary links with a capacity of 100 Mbit/s. Obviously the most utilized link is link number 2 with a total utilization of 80 percent. The green fraction of the bar indicates that 40 Mbit/s are going to be re-distributed into the MPLS network.

Since we already decided which flows to transfer to the MPLS overlay network we of course know which flows will remain in the initial OSPF network. The next step take the remaining network and regard it as a new independent network in itself. The remaining flows represent the demand matrix in our new network. Figure 2 already anticipates the result. Note that the green traffic is obviously distributed more equally among the remaining capacities now. In our final step we are going to combine the two networks.

Figure 3 shows the final utilizations for the four links taken out of our hypothetical example. When comparing the result to the initial Figure 1 we notice a significant improvement of the network congestion at first sight. The traffic itself is now more evenly distributed among the four links. Apparently, the more flows we take out of the OSPF network (i.e.

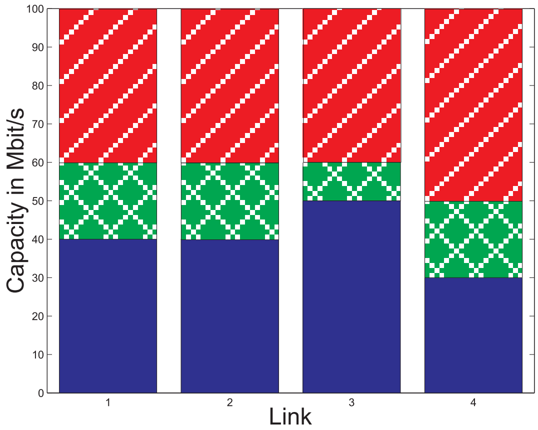


Figure 3: The combined network

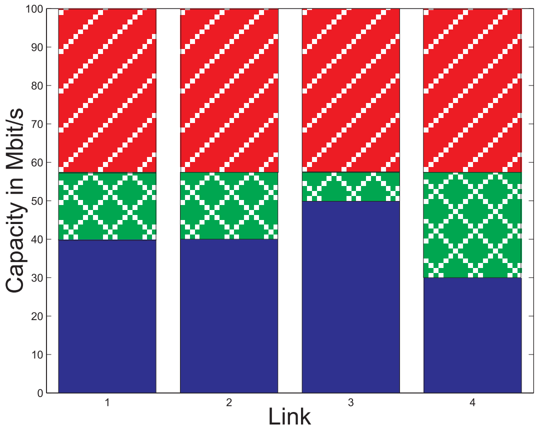
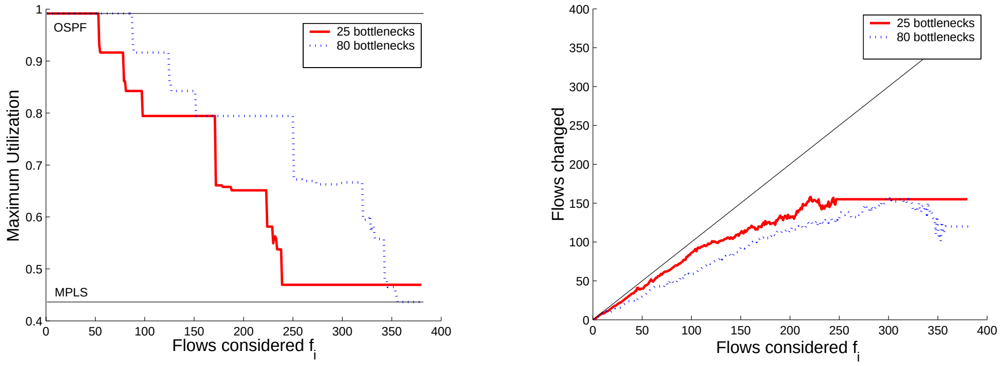


Figure 4: Optimized pure MPLS network

the more green traffic we have) the closer we can get to the lower bound introduced with the pure MPLS network in Figure 4. To investigate the convergence behavior to the lower bound in more detail, we focus on the two largest networks presented in the appendix.

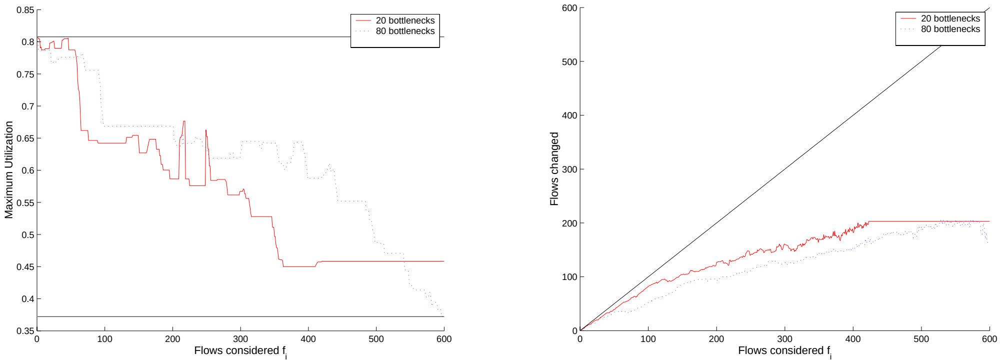


(a) Maximum Utilization

(b) Changed flows

Figure 5: Considering 25 and 80 bottlenecks in the 20 router network

The 20 node network has a total of 102 links and 380 flows. Figure 5(a) summarizes the results for $b = 25$ and $b = 80$ bottlenecks in dependency of the considered flows f_i. As shown in the graph the lines are not decreasing strictly monotonic, which leads to the conclusion that re-distributing more flows do not necessarily lead to better results in the short-term. The redistributed flows are dependent on the underlying OSPF network, the number of considered flows and the sorting defined as part of the algorithm. Figure 5(b) plots the ratio of the considered flows f_i to the changed flows f_o. Besides the number of considered flows, the speed of convergence to the optimum strong depend on the number of bottlenecks taken into account. A higher number of bottlenecks considered leads to to a slower convergence but increases the probability to reach the optimum. For $b = 20$ in Figure 5(a) a limit value above the optimum is observed. The limit is founded in the restricted number of flows for $b = 20$. To reach the optimum, flows which are not traversing any of the twenty bottlenecks have to be redirected. In the case of $b = 80$ the number of investigated bottlenecks is higher and the pure MPLS solution is achieved.



(a) Maximum Utilization

(b) Flows

Figure 6: Considering 20 and 80 bottlenecks in the 25 router network

The similar shape of the curves for $b = 25$ and $b = 80$ in Figure 5(a) is accidentally, as can be seen in the figures for the 25 node network. Here the shape for $b = 20$ and $b = 80$ differs considerable. Basically, the analysis of this network confirms the evaluations and results of the 20 node network. Based on our results and dependent on the size of the network, we therefore suggest to consider between 20 and 50 percent of all links as bottlenecks and 40 to 60 percent of all the flows.

4.2 Simplifying a MPLS network with the use of OSPF

A special case of the DDR-algorithm is the possibility to simplify a MPLS network with the use of OSPF. If f_i is equal to the total number of flows and b to total the number of links, the algorithm is operating like our MPLS optimization algorithm. The only difference is that in the last step the paths of the MPLS flows are compared to the paths of the OSPF flows. If they match each other, one does not need to set up a MPLS path but can still use the existing OSPF path. The average and maximum utilization are equal to the MPLS approach and the only difference consists in the number of MPLS pathes you have to set up. Because of this, we refer to the results and reference points of the MPLS algorithm to verify the performance of the algorithm. In this special case, the number of flows that are not following the original OSPF path are of interest.

Networks	10		14		18		20		25	
	Def. OSPF	L.O. OSPF	Def. OSPF	L.O. OSPF	Def. OSPF	L.O. OSPF	Def. OSPF	L.O. OSPF	Def. OSPF	L.O. OSPF
Number of flows	22	8	47	20	54	113	119	122	164	150
Percentage of flows	24.4	8.9	25.8	11.0	17.6	36.9	31.3	32.1	27.3	25.0

Table 6: Number and percentage of changed flows

There may be other reasons like QoS to set up more MPLS pathes. However from the performance point of view, less than 40% of the flows have to be configured as MPLS paths - see Table 6 and also Figure 5(b) and 6(b).

4.3 Global vs. local Traffic Engineering

Based on the results presented in the paper we are able to offer valuable clues to decide about the preferred use of a local or a global TE system. Local TE concentrates on local components of the global problem. Global Traffic Engineering on the other hand is intended to re-distribute the traffic from scratch. Figure 7 illustrates this behavior.

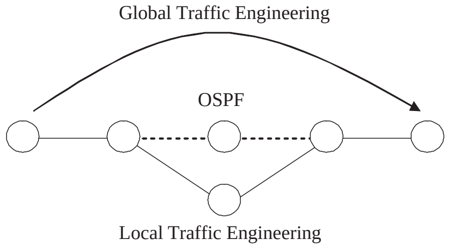


Figure 7: Local and global Traffic Engineering

While local TE attempts to route the flow around the bottleneck global TE is able to look for a completely new path from the source to the destination. In principle, global Traffic Engineering thus offers greater potential. When considering TE we must not loose track of the average delay. That is, besides the maximum and average utilization we also have to pay close attention to the average delay when using completely new paths for the flows of the MPLS overlay network. In a first approximation we therefore define the delay of a flow as the number of hops used. Table 7 shows the increase in the average delay. While the first row states the number of changes made by each individual approach the rows two and three describe the number of additional hops per change (AHPC) respectively the number of additional hops per flow (AHPF).

Network	10	14	18	20	25
Changes	22	47	54	122	119
AHPC	0.44	0.63	0.34	0.77	0.77
AHPF	1.82	2.44	1.94	2.39	2.45

Table 7: Increase in the delay

With regard to all flows we only face an additional delay smaller than 2.5 hops per flow. In the first view, this leads to the conclusion that only small changes like bypassing the bottleneck are made. To decide, if it is possible to take advantage of this behavior by a local Traffic Engineering approach, we have to take a closer look at the path distribution of the chosen paths.

If the new path distribution only bypasses the bottlenecks, we could easily develop a local Traffic Engineering approach. Otherwise, it will be complicated to construct a simple and local Traffic Engineering approach, as it will be very difficult to decide which particular flows to change and how to do so. That is, a local Traffic Engineering system would have to find an appropriate set of flows traversing the bottleneck link and subsequently signal the individual flows when and where to change their original path.

Hops	0	1	2	3
18 node network	45	33/7	7/12	1/5
20 node network	7	20/14	6/6	0/0
25 node network	30	52/38	14/18	1/5

Table 8: Number of hops that the perfect path differs from the original path before/after the bottleneck.

However, there is no general way to do so. The number of hops in Table 8 describes the distance from the bottleneck that the optimized path differs from the original path in hops. As shown in the Table in the ideal solution the flows tend to avoid the bottleneck, leaving and rejoining their original path an indeterminate number of hops before and after the actual bottleneck link. Moreover, the changed paths of a considerable number of flows even still include the bottleneck link (hops = 0). A behavior that clearly indicates the need for global Traffic Engineering, as it is almost impossible to come to a local decision of how to place the individual flows to finally reach the quality of the global optimization.

5 Conclusions and Outlook

In this paper we investigated possibility of combining common network technologies like IGP and MPLS to a Traffic Engineering system. We introduced a fast linear optimization approach for MPLS networks, with a remarkable objective function for TE and compared default, optimized OSPF routing and MPLS TE and to reveal the advantages of MPLS TE. The proposed overlay model consisting of a mix of the OSPF and the MPLS routing scheme reduces the state space and offers simplified configuration for network operators. In general, our routing optimization works well for a given traffic distribution. However, the real internet traffic is not static but varies over time. The quality of our result, applied to varying traffic demands, still has to be investigated. This leads to the question of how to compare two different routing schemes in general and also of how to evaluate the quality of a routing decision. In our work the results were compared mainly by the average and maximum link utilizations of sample networks. To deploy a more general approach to evaluate routing strategies independent from the network would be a fascinating but difficult task. And last but not least, the influence of different traffic classes and QoS would be an interesting point to investigate in detail, as well.

References

[1] Awduche, D., Malcolm, J., Agogbua, J., O'Dell, M. and J. McManus, *Requirements for Traffic Engineering Over MPLS*, RFC 2702, September 1999

[2] X. Xiao, A. Hannan, B. Bailey, and L. Ni, *Traffic Engineering with MPLS in the Internet*, IEEE Network, March/April 2000

[3] LTM Berry, S. Köhler, D. Staehle and P. Tran-Gia, *Fast heuristic for optimal routing in large IP networks*, Report No.262, July 2000

[4] S. Köhler, D. Staehle and U. Kohlhaas, *Optimization of IP Routing by Link Cost Specification*, 15th ITC Specialist Seminar, July 2002, Würzburg, Germany

[5] B. Fortz and M. Thorup, *Optimizing OSPF/IS-IS Weights in a Changing World*, IEEE Journal on Selected Areas in Communications, Vol. 20, No. 4, May 2002

[6] B. Fortz and M. Thorup, *Increasing Internet Capacity Using Local Search*, Technical report, AT&T Labs Research, August 2000

[7] Moy, *OSPF Version 2*, Request for Comments 2328, April 1998

[8] Rosen, et al., *Multiprotocol Label Switching Architecture*, Request for Comments 3031, Januar 2001

[9] Awduche, et. al., *Overview and Principles of Internet TE*, Request for Comments 3272, May 2002

[10] Crawley, et. al., *A Framework for QoS-based Routing in the Internet*, RFC 2386, August 1998

[11] ILOG, Inc., ILOG CPLEX, *http://www.ilog.com/products/cplex/*, 1987-2002.

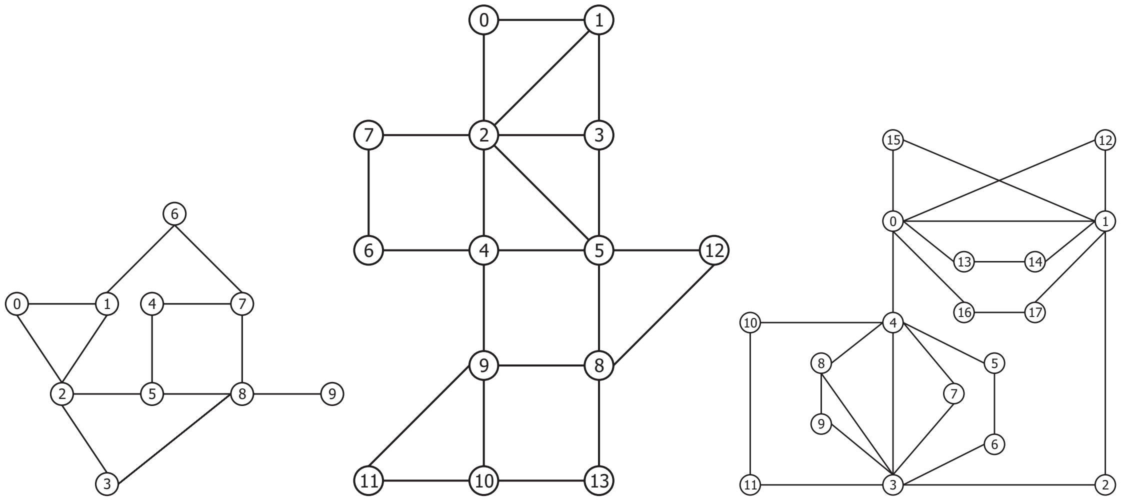
[12] C. Huitema, *Routing in the Internet*, Second Edition, Prentice Hall, 1999

[13] G. Apostolopoulos et al, *QoS Routing Mechanisms and OSPF Extensions*, draft-guerin-qos-routing-ospf-04, December, 1998

[14] A. Bley, M. Grötschel, R. Wessäly, *Design of Broadband Virtual Private Networks: Model and Heuristics for the B-WiN*, Konrad-Zuse-Zentrum fr Informationstechnik Berlin (ZIB), Preprint SC-98-13, 1998

Appendix

In the following a presentation of the visualization and the matrices representing the capacities and demands of five sample networks is given.



(a) 10 node network (b) 14 node network (c) 18 node network

Figure 8: The 10, 14 and 18 node networks

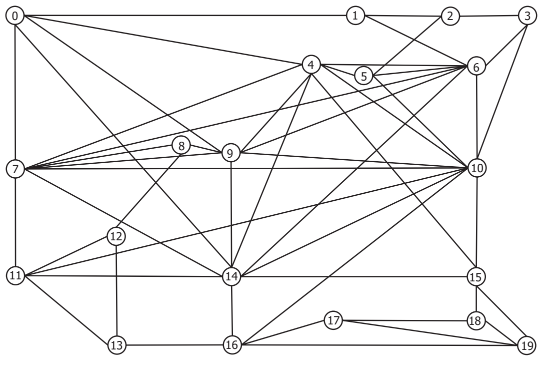


Figure 9: 20 node network

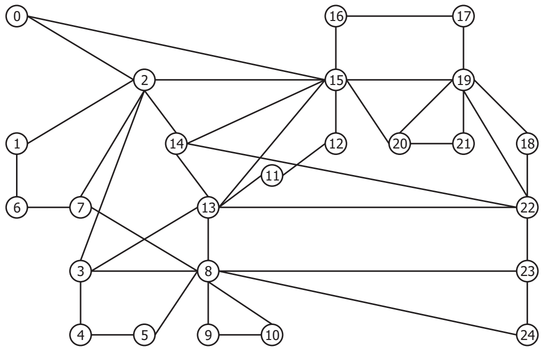
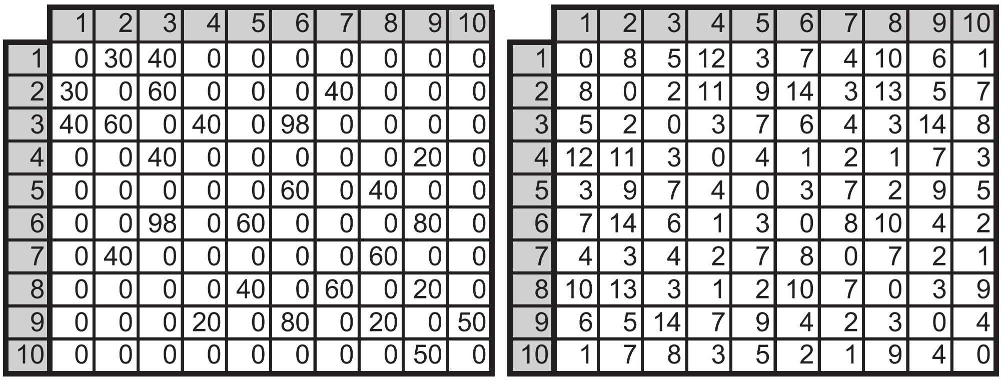


Figure 10: 25 node network



	1	2	3	4	5	6	7	8	9	10
1	0	30	40	0	0	0	0	0	0	0
2	30	0	60	0	0	0	40	0	0	0
3	40	60	0	40	0	98	0	0	0	0
4	0	0	40	0	0	0	0	0	20	0
5	0	0	0	0	0	60	0	40	0	0
6	0	0	98	0	60	0	0	0	80	0
7	0	40	0	0	0	0	0	60	0	0
8	0	0	0	0	40	0	60	0	20	0
9	0	0	0	20	0	80	0	20	0	50
10	0	0	0	0	0	0	0	0	50	0

(a) capacities

	1	2	3	4	5	6	7	8	9	10
1	0	8	5	12	3	7	4	10	6	1
2	8	0	2	11	9	14	3	13	5	7
3	5	2	0	3	7	6	4	3	14	8
4	12	11	3	0	4	1	2	1	7	3
5	3	9	7	4	0	3	7	2	9	5
6	7	14	6	1	3	0	8	10	4	2
7	4	3	4	2	7	8	0	7	2	1
8	10	13	3	1	2	10	7	0	3	9
9	6	5	14	7	9	4	2	3	0	4
10	1	7	8	3	5	2	1	9	4	0

(b) demands

Figure 11: Capacity and demand matrix for the 10 router network

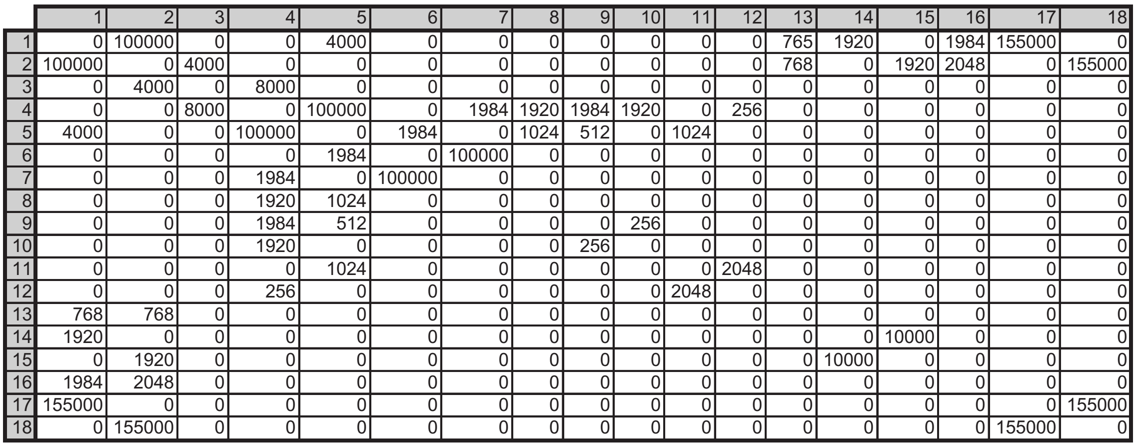
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	0	110	90	0	0	0	0	0	0	0	0	0	0	0
2	90	0	105	107	0	0	0	0	0	0	0	0	0	0
3	98	87	0	78	89	88	0	108	0	0	0	0	0	0
4	0	102	105	0	0	89	0	0	0	0	0	0	0	0
5	0	0	100	0	0	90	95	0	0	103	0	0	0	0
6	0	0	86	110	103	0	0	0	120	0	0	0	100	0
7	0	0	0	0	110	0	0	90	0	0	0	0	0	0
8	0	0	93	0	0	0	88	0	0	0	0	0	0	0
9	0	0	0	0	0	103	0	0	0	111	0	0	106	112
10	0	0	0	0	104	0	0	0	112	0	107	109	0	0
11	0	0	0	0	0	0	0	0	0	104	0	106	0	107
12	0	0	0	0	0	0	0	0	0	113	115	0	0	0
13	0	0	0	0	0	102	0	0	120	0	0	0	0	0
14	0	0	0	0	0	0	0	0	100	0	105	0	0	0

(a) capacities

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	0	2	3	2	3	3	2	3	2	3	2	3	2	3
2	1	0	2	1	2	1	2	2	1	1	2	1	2	2
3	2	3	0	1	2	3	1	2	3	2	1	2	3	1
4	1	2	3	0	1	2	2	2	1	1	1	3	2	1
5	1	2	1	3	0	2	3	1	2	2	1	1	2	3
6	3	2	2	2	1	0	2	1	1	2	1	1	2	1
7	1	3	2	1	2	1	0	1	2	2	3	3	1	1
8	1	2	1	3	1	1	1	0	2	3	3	3	2	3
9	2	1	2	3	2	1	2	3	0	1	2	2	3	3
10	1	2	3	1	2	3	1	2	3	0	3	2	1	2
11	1	2	2	3	2	1	2	3	2	1	0	2	3	3
12	3	2	2	3	2	1	2	3	1	1	2	0	3	2
13	1	2	3	2	2	2	3	2	1	2	3	2	0	2
14	2	2	2	3	2	1	1	2	3	2	1	2	3	0

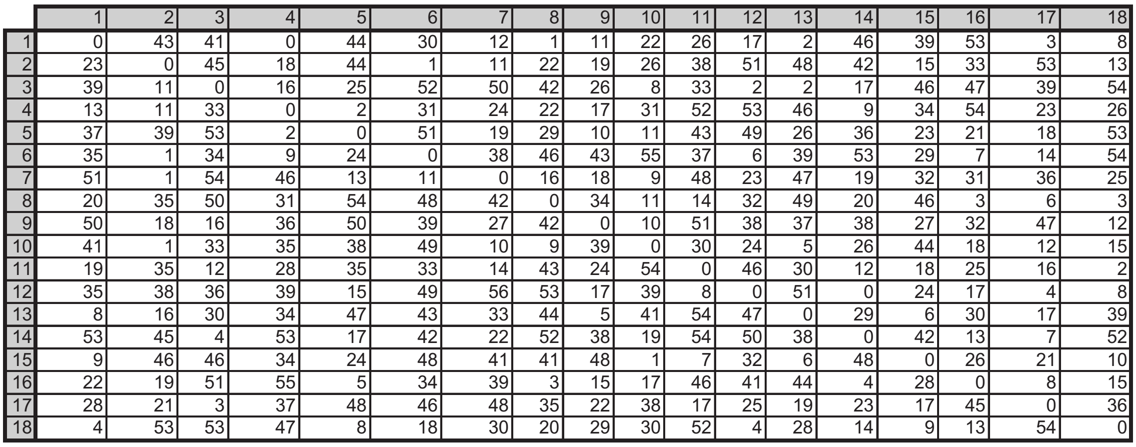
(b) demands

Figure 12: Capacity and demand matrix for the 14 router network



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	0	100000	0	0	4000	0	0	0	0	0	0	0	765	1920	0	1984	155000	0
2	100000	0	4000	0	0	0	0	0	0	0	0	0	768	0	1920	2048	0	155000
3	0	4000	0	8000	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	8000	0	100000	0	1984	1920	1984	1920	0	256	0	0	0	0	0	0
5	4000	0	0	100000	0	1984	0	1024	512	0	1024	0	0	0	0	0	0	0
6	0	0	0	0	1984	0	100000	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	1984	0	100000	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1920	1024	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	1984	512	0	0	0	0	256	0	0	0	0	0	0	0	0
10	0	0	0	1920	0	0	0	0	256	0	0	0	0	0	0	0	0	0
11	0	0	0	0	1024	0	0	0	0	0	0	2048	0	0	0	0	0	0
12	0	0	0	256	0	0	0	0	0	0	2048	0	0	0	0	0	0	0
13	768	768	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	1920	0	0	0	0	0	0	0	0	0	0	0	0	0	10000	0	0	0
15	0	1920	0	0	0	0	0	0	0	0	0	0	0	10000	0	0	0	0
16	1984	2048	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	155000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	155000
18	0	155000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	155000	0

(a) capacities



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	0	43	41	0	44	30	12	1	11	22	26	17	2	46	39	53	3	8
2	23	0	45	18	44	1	11	22	19	26	38	51	48	42	15	33	53	13
3	39	11	0	16	25	52	50	42	26	8	33	2	2	17	46	47	39	54
4	13	11	33	0	2	31	24	22	17	31	52	53	46	9	34	54	23	26
5	37	39	53	2	0	51	19	29	10	11	43	49	26	36	23	21	18	53
6	35	1	34	9	24	0	38	46	43	55	37	6	39	53	29	7	14	54
7	51	1	54	46	13	11	0	16	18	9	48	23	47	19	32	31	36	25
8	20	35	50	31	54	48	42	0	34	11	14	32	49	20	46	3	6	3
9	50	18	16	36	50	39	27	42	0	10	51	38	37	38	27	32	47	12
10	41	1	33	35	38	49	10	9	39	0	30	24	5	26	44	18	12	15
11	19	35	12	28	35	33	14	43	24	54	0	46	30	12	18	25	16	2
12	35	38	36	39	15	49	56	53	17	39	8	0	51	0	24	17	4	8
13	8	16	30	34	47	43	33	44	5	41	54	47	0	29	6	30	17	39
14	53	45	4	53	17	42	22	52	38	19	54	50	38	0	42	13	7	52
15	9	46	46	34	24	48	41	41	48	1	7	32	6	48	0	26	21	10
16	22	19	51	55	5	34	39	3	15	17	46	41	44	4	28	0	8	15
17	28	21	3	37	48	46	48	35	22	38	17	25	19	23	17	45	0	36
18	4	53	53	47	8	18	30	20	29	30	52	4	28	14	9	13	54	0

(b) demands

Figure 13: Capacity and demand matrix for the 18 router network

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	0	100	0	0	100	0	0	100	0	100	0	0	0	0	100	0	0	0	0	0
2	100	0	100	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	100	0	100	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	100	0	0	0	100	0	0	0	100	0	0	0	0	0	0	0	0	0
5	100	0	0	0	0	100	100	100	0	100	100	0	0	0	100	100	0	0	0	0
6	0	0	100	0	100	0	100	0	0	0	100	0	0	0	0	0	0	0	0	0
7	0	100	0	100	100	100	0	100	0	100	100	0	0	0	100	0	0	0	0	0
8	100	0	0	0	100	0	100	0	100	100	100	100	0	0	100	0	0	0	0	0
9	0	0	0	0	0	0	0	100	0	100	0	0	100	0	0	0	0	0	0	0
10	100	0	0	0	100	0	100	100	100	0	100	0	0	0	100	0	0	0	0	0
11	0	0	0	100	100	100	100	100	0	100	0	100	0	0	100	100	100	0	0	0
12	0	0	0	0	0	0	0	100	0	0	100	0	100	100	100	0	0	0	0	0
13	0	0	0	0	0	0	0	0	100	0	0	100	0	100	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	100	100	0	0	0	100	0	0	0
15	100	0	0	0	100	0	100	100	0	100	100	100	0	0	0	100	100	0	0	0
16	0	0	0	0	100	0	0	0	0	0	100	0	0	0	100	0	0	0	100	100
17	0	0	0	0	0	0	0	0	0	0	100	0	0	100	100	0	0	100	0	100
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	100	100
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	100	0	100
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	100	100	100	0

(a) capacities

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	0.00	2.71	0.68	1.97	4.79	1.30	5.39	1.00	1.22	1.03	2.85	5.51	0.89	1.88	2.04	2.38	2.42	0.77	0.95	1.30
2	2.80	0.00	1.36	3.95	9.60	2.61	10.80	2.01	2.45	2.06	5.71	11.04	1.78	3.77	4.08	4.77	4.84	1.55	1.91	2.61
3	0.67	1.29	0.00	0.94	2.28	0.62	2.57	0.48	0.58	0.49	1.36	2.63	0.42	0.90	0.97	1.13	1.15	0.37	0.45	0.62
4	2.00	3.88	0.97	0.00	6.86	1.87	7.72	1.44	1.75	1.47	4.08	7.89	1.27	2.70	2.92	3.41	3.46	1.11	1.36	1.87
5	5.22	10.12	2.53	7.37	0.00	4.87	20.15	3.74	4.56	3.84	10.65	20.59	3.32	7.04	7.61	8.90	9.04	2.89	3.56	4.87
6	1.30	2.52	0.63	1.83	4.46	0.00	5.02	0.93	1.14	0.96	2.65	5.13	0.83	1.75	1.90	2.21	2.25	0.72	0.89	1.21
7	5.98	11.58	2.90	8.43	20.48	5.57	0.00	4.28	5.22	4.40	12.19	23.56	3.80	8.05	8.71	10.18	10.34	3.31	4.07	5.58
8	0.99	1.93	0.48	1.40	3.40	0.93	3.83	0.00	0.87	0.73	2.03	3.92	0.63	1.34	1.45	1.69	1.72	0.55	0.68	0.93
9	1.22	2.36	0.59	1.72	4.17	1.13	4.69	0.87	0.00	0.89	2.48	4.80	0.77	1.64	1.77	2.07	2.10	0.67	0.83	1.14
10	1.02	1.98	0.49	1.44	3.49	0.95	3.93	0.73	0.89	0.00	2.08	4.02	0.65	1.37	1.49	1.74	1.76	0.57	0.69	0.95
11	2.96	5.73	1.43	4.17	10.13	2.75	11.40	2.12	2.58	2.17	0.00	11.66	1.88	3.98	4.31	5.03	5.11	1.64	2.01	2.76
12	6.13	11.88	2.97	8.65	21.00	5.71	23.64	4.39	5.35	4.51	12.49	0.00	3.90	8.25	8.93	10.44	10.60	3.40	4.17	5.72
13	0.88	1.70	0.43	1.24	3.01	0.82	3.39	0.63	0.77	0.65	1.79	3.47	0.00	1.18	1.28	1.50	1.52	0.49	0.60	0.82
14	1.91	3.70	0.92	2.69	6.53	1.78	7.35	1.37	1.67	1.40	3.89	7.52	1.21	0.00	2.78	3.25	3.30	1.06	1.30	1.78
15	2.07	4.01	1.00	2.92	7.10	1.93	7.99	1.48	1.81	1.52	4.22	8.16	1.32	2.79	0.00	3.53	3.58	1.15	1.41	1.93
16	2.44	4.73	1.18	3.44	8.36	2.27	9.41	1.75	2.13	1.80	4.98	9.62	1.55	3.29	3.56	0.00	4.22	1.35	1.66	2.28
17	2.48	4.81	1.20	3.50	8.50	2.31	9.57	1.78	2.17	1.83	5.06	9.78	1.58	3.34	3.62	4.23	0.00	1.38	1.69	2.32
18	0.76	1.48	0.37	1.08	2.62	0.71	2.95	0.55	0.67	0.56	1.56	3.01	0.49	1.03	1.11	1.30	1.32	0.00	0.52	0.71
19	0.94	1.83	0.46	1.33	3.23	0.88	3.64	0.68	0.82	0.69	1.92	3.72	0.60	1.27	1.37	1.61	1.63	0.52	0.00	0.88
20	1.30	2.52	0.63	1.84	4.46	1.21	5.02	0.93	1.14	0.96	2.66	5.13	0.83	1.75	1.90	2.22	2.25	0.72	0.89	0.00

(b) demands

Figure 14: Capacity and demand matrix for the 20 router network

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
1	0	0	2480	0	0	0	0	0	0	0	0	0	0	0	0	2480	0	0	0	0	0	0	0	0	0
2	0	0	2480	0	0	0	2480	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	2480	2480	0	2480	0	0	0	2480	0	0	0	0	0	0	2480	2480	0	0	0	0	0	0	0	0	0
4	0	0	2480	0	2480	0	0	0	2480	0	0	0	0	2480	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	2480	0	2480	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	2480	0	0	0	2480	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	2480	0	0	0	0	0	2480	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	2480	0	0	0	2480	0	2480	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	2480	0	2480	0	2480	0	2480	2480	0	0	2480	0	0	0	0	0	0	0	0	0	2480	2480
10	0	0	0	0	0	0	0	0	2480	0	2480	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	2480	2480	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	2480	2480	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	2480	0	0	0	2480	0	0	0	0	0	0	0	0	0
14	0	0	0	2480	0	0	0	0	2480	0	0	2480	0	0	2480	2480	0	0	0	0	0	0	2480	0	0
15	0	0	2480	0	0	0	0	0	0	0	0	0	0	2480	0	2480	0	0	0	0	0	0	2480	0	0
16	2480	0	2480	0	0	0	0	0	0	0	0	0	2480	2480	2480	0	2480	0	0	2480	2480	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2480	0	2480	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2480	0	0	2480	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2480	0	0	2480	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2480	0	2480	2480	0	9920	2480	2480	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2480	0	0	0	9920	0	2480	0	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2480	2480	0	0	0	0
23	0	0	0	0	0	0	0	0	0	0	0	0	0	2480	2480	0	0	0	2480	2480	0	0	0	2480	0
24	0	0	0	0	0	0	0	0	2480	0	0	0	0	0	0	0	0	0	0	0	0	0	2480	0	2480
25	0	0	0	0	0	0	0	0	2480	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2480	0

(a) capacities

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
1	0	46	18	11	6	52	2	48	37	11	49	25	15	31	53	42	64	57	45	43	19	0	52	53	42
2	56	0	26	1	37	47	26	44	6	65	44	40	62	48	56	62	11	60	19	17	32	21	63	9	23
3	58	10	0	5	37	31	39	19	50	63	1	10	41	34	46	49	49	7	60	31	41	57	3	3	6
4	40	7	56	0	4	64	33	2	56	33	50	9	43	19	2	65	58	9	12	22	39	27	11	64	59
5	60	43	4	31	0	27	20	13	11	6	1	63	51	13	28	40	60	48	50	27	9	65	42	37	18
6	57	50	9	3	2	0	9	43	58	38	45	14	16	21	64	36	6	28	64	61	57	34	24	11	55
7	57	34	64	63	63	39	0	47	6	56	54	43	57	53	37	7	65	34	65	18	9	21	7	23	33
8	1	9	39	64	16	42	3	0	56	32	38	30	32	21	48	10	28	9	38	34	30	63	7	46	26
9	29	44	11	46	63	22	53	62	0	32	50	25	10	32	13	42	7	27	42	2	34	65	37	42	17
10	65	24	60	36	29	1	54	12	23	0	1	18	61	35	0	13	54	53	19	15	57	8	43	21	41
11	52	37	24	15	45	16	63	31	66	47	0	33	19	4	18	58	23	33	30	38	13	9	6	55	34
12	11	21	38	64	52	19	36	23	3	23	45	0	13	56	38	61	58	2	20	36	17	23	29	22	52
13	31	37	36	20	42	15	3	16	5	40	28	49	0	62	1	52	10	57	11	50	28	52	3	26	11
14	14	64	51	29	41	53	43	21	4	7	15	30	46	0	48	13	12	21	5	20	56	54	28	34	51
15	15	23	56	22	33	49	17	63	63	45	33	12	2	18	0	25	42	28	64	9	16	48	34	44	26
16	63	50	18	25	27	50	19	58	24	23	51	46	56	32	26	0	50	0	39	36	20	29	27	1	24
17	62	14	23	46	63	15	60	55	22	59	59	54	32	61	42	3	0	56	54	58	51	27	44	51	1
18	20	65	16	65	34	28	11	2	19	4	33	14	25	8	24	26	4	0	57	22	5	42	30	12	49
19	32	37	4	10	38	27	25	1	9	23	32	53	49	14	29	4	28	57	0	48	42	51	6	35	56
20	4	13	13	28	13	38	35	24	25	33	46	57	34	30	12	2	2	17	46	0	53	24	21	30	52
21	40	11	46	13	46	49	61	58	34	38	45	39	19	10	21	13	60	50	48	23	0	12	14	16	11
22	55	34	54	24	24	61	59	30	22	41	41	34	32	20	7	7	58	57	33	45	55	0	46	18	28
23	49	38	43	30	20	48	40	33	43	1	22	20	36	58	30	14	29	7	14	42	17	43	0	42	65
24	60	3	58	15	34	26	5	11	54	26	29	27	26	49	27	65	57	56	16	57	34	46	57	0	8
25	64	38	51	2	60	22	10	63	43	19	41	10	53	4	31	28	24	51	42	58	48	39	49	30	0

(b) demands

Figure 15: Capacity and demand matrix for the 25 router network