MPLS Traffic Engineering in OSPF Networks - A combined approach

Stefan Köhler and Andreas Binzenhöfer
Dept. of Distributed Systems, Inst. of Computer Science, Univ. of Würzburg, Germany
E-Mail: [koehler, binzenhoefer]@informatik.uni-wuerzburg.de

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 TE of their networks can have a tactical and strategic value [2]. The first part of the paper introduces a linear optimization approach for TE in MPLS networks. Most remarkable on the presented approach in comparison to other linear approaches 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 shows the benefit of OSPF optimization and MPLS TE. Based on this comparison we can offer some valuable clues to decide in favor of a local or global TE system.

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 (TE) in IP networks. TE is the process of controlling the way traffic flows through a network to optimize resource utilization and network performance. TE 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.

Besides the simple configuration aspect of the shortest path, this principle conserves network resources. However, it also causes the problem that the shortest paths from different sources overlap at some links, resulting in congestion on those links and unequally distributed traffic in the network.This possibly leads to situations where the traffic from source to destination is blocked, while a longer path between these two routers is underutilized.

To provide improved support for TE, the IETF introduced MPLS [8], Constraint-base Routing [10], enhanced link state IGPs [13] and a modification of the IGP metric [9].

MPLS avoids the problems of IGP by extending the way a path from source to destination is determined 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 IGP Routing approaches to a common TE system and deals with the corresponding evaluations and results. Finally, Chapter 5 concludes the paper with a summary and an outlook.

2. Optimization problem formulation

Unlike IGP routing algorithms such as 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 $N \times N$ 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)

A linear program (LP) in principle consists of two parts. An objective function and linear constraints consisting of equations and in-equations. The corresponding objective function to minimize the maximum and the average utilization is shaped as follows:

$$a_t t + \frac{1}{L} \sum_{(i,j)} \sum_{uv} \frac{f^{uv} x_{ij}^{uv}}{c_{ij}} \;, \quad c_{ij} \neq 0 \qquad (1)$$

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 a real number between 0 and 1. It defines the percentage of the flow $f^{uv} \geq 0$ routed through link (i, j). The variable L specifies the total number of links in the network. 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 limit is specified by a_c. If $a_t \neq 0$ then a_c is set to the maximum utilization t.

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

- Transport constraints: The transport constraints ensure that a flow takes a well defined path(s) through the network. A flow from node u to node v has to originate from the source u to its next hop(s):

$$\sum\nolimits_{i=1,c_{ui}>0}^{N} x_{ui}^{uv} - \sum\nolimits_{i=1,c_{iu}>0}^{N} x_{iu}^{uv} \geq 1, \quad i \neq u \tag{3}$$

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

$$\sum\nolimits_{j=1,c_{ij}>0}^{N} x_{ij}^{uv} - \sum\nolimits_{j=1,c_{ji}>0}^{N} x_{ji}^{uv} = 0 \quad \forall i \notin \{u,v\} \tag{4}$$

  Finally each flow must reach a destination router v:

$$\sum\nolimits_{i=1,c_{iv}>0}^{N} x_{iv}^{uv} - \sum\nolimits_{i=1,c_{vi}>0}^{N} x_{vi}^{uv} \geq 1, \quad i \neq v \tag{5}$$

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 dependend $N \times N$ 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 simultaneously optimizing the maximum utilization and the average utilization. Other optimization approaches ([5], [14]) are mainly concentrating on one part of the objective function used in this paper. The chosen objective function fits the definition of TE given at the beginning of the paper. The minimization of the maximum link utilization complies with the performance principle and the reduction of the average utilization copes with the network resources. Table 1 illustrates the differences between default OSPF and linear optimization of MPLS networks focusing on average utilization, maximum utilization and finally on both at once. The computation times 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 [15]. Using only a part of the proposed objective function does not lead to the desired results. The optimization of the average utilization of MPLS networks and the inherent shortest path principle of OSPF lead to similar results. The small differences occur as we use default Cisco weights instead of a simple hop count metric. Reducing the maximum utilization only, resulted in a very high average

Table 1
Linear optimization in the 20 and 25 node network

Network	20				25			
	OSPF	Avg.	Max.	Avg. & Max.	OSPF	Avg.	Max.	Avg. & Max.
Maximum	0.992	0.992	0.436	0.436	0.808	0.783	0.372	0.372
Average	0.234	0.234	0.417	0.238	0.255	0.255	0.334	0.262

Table 2
Number of total/multiple splits.

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

network utilization and did thus not sufficiently reduce the network congestion. When applying linear optimization to our sample networks using the MPLS forwarding scheme the maximum utilization was reduced to the assumed optimum. The splits do not appear in the expected amount (see Table 2) even though all variables had been declared as real numbers. 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 if 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

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

to compare our solution with a default configured (default CISCO OSPF configuration [2]) and a weight optimized OSPF network [4]. Table 3 compares the unoptimized and optimized OSPF results to our new lower bound found by linear optimization of the MPLS networks. Note that the largest improvement is between default and optimized OSPF.

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

Network	10			20			25		
	Def. OSPF	L.O. OSPF	MPLS	Def. OSPF	L.O. OSPF	MPLS	Def. OSPF	L.O. OSPF	MPLS
Minimum	0	0.65	0.603	0.007	0.007	0.007	0.001	0.001	0.001
Maximum	1.296	0.967	0.942	0.992	0.530	0.436	0.808	0.480	0.372
Average	0.817	0.829	0.827	0.234	0.233	0.238	0.255	0.256	0.262

This justifies the optimization of OSPF weights. Furthermore, MPLS networks are able to reduce the network congestion beyond the possibilities of pure OSPF networks.

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

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. Thus, MPLS obviously offers greater potential for the optimization of larger networks.

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

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

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 in doing so is, that it greatly extends the possibilities for TE and offers new ways to reduce the number of MPLS paths. 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 optimally 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 find out how to combine MPLS and OSPF networks. This section will therefore deal with the following three steps:

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

To distinguish the following algorithm from the pure MPLS optimization approach we name the concept D^2R (Decompose-Design-Re-assemble)-algorithm.

Decompose a network

We start off 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 fast ways to optimize networks our following results will be based 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 the 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. Replacing flows with a very large bandwidth proved to be too inflexible as that way the problem area was simply shifted to another location. Flows carrying a small bandwidth, however, 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 sort this set of flows by their bandwidth in ascending order.

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 bottlenecks are sorted by the second criterion, their bandwidth, in ascending order. 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. Furthermore, we want to emphasize that the parameter f_i determines

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

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

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 including it into our MPLS overlay network respectively.

Design the MPLS overlay network

The next step is building the MPLS overlay network. In the following figures the striped bars are used to denote the unused capacities, the plain bars stand for the bandwidth used by flows of the OSPF network and the checkered bars describe the bandwidth used by flows that will be re-distributed into 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 checkered fraction of the bar indicates that 40 Mbit/s are going to be re-distributed into the MPLS network. In the next step we take the remaining network and regard it as a new independent network in itself. The remaining

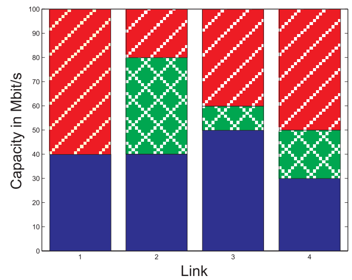


Figure 1. Utilizations in the initial OSPF network

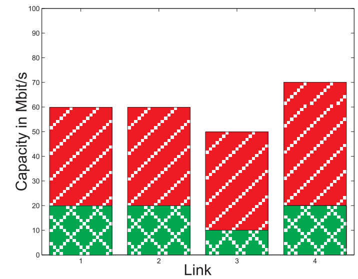


Figure 2. Optimized utilizations for the MPLS overlay network

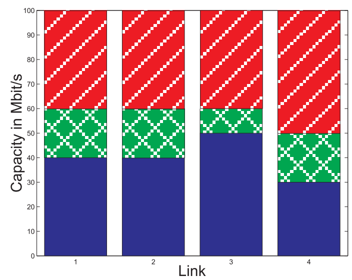


Figure 3. The combined network

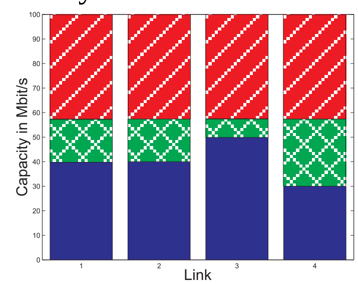
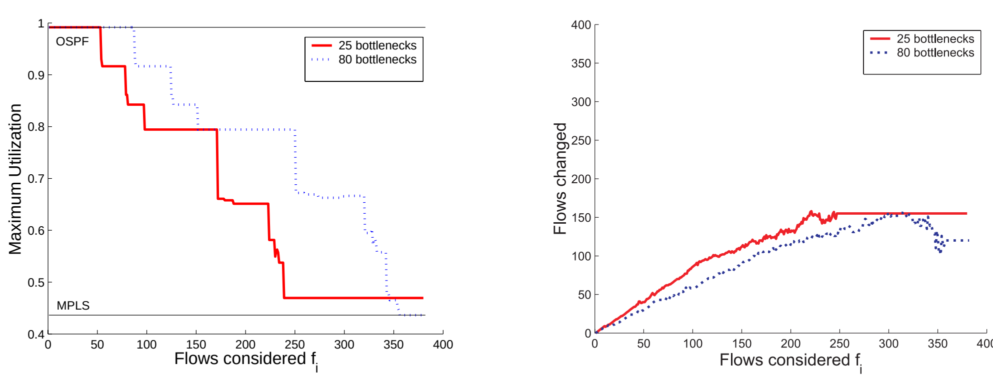


Figure 4. Optimized pure MPLS network

flows represent the demand matrix in our new network. Figure 2 already anticipates the result. Note that the checkered 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 of 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. the more checkered traffic we have) the closer we can get to the lower bound found by linear optimization of the MPLS network as shown in Figure 4. To analyze the convergence behavior to the lower bound in more detail, we focus on the two largest networks presented in [15]. The 20 node network has a total of 102 links and 380 flows. Figure

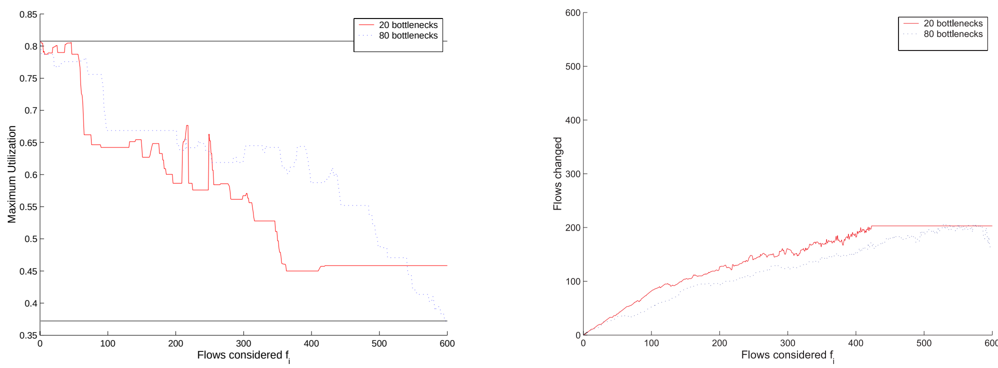


(a) Maximum Utilization

(b) Changed flows

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

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 strictly monotonic decreasing, which leads to the conclusion that redistributing more flows does not necessarily lead to better results in the short-run. Which flows will be redistributed depends on the underlying OSPF network, the number of flows considered and the sorted list of flows. 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 strongly depends on the number of bottlenecks taken into account. A higher number of bottlenecks considered leads to a slower convergence but increases the probability to reach the optimum. For $b = 25$ in Figure 5(a) the curve does not reach the optimum. This is due to the fact that flows which are not traversing any of the 25 bottlenecks would have to be redistributed to reach the optimum. In the case of $b = 80$ the number of investigated bottlenecks is large enough to attain the pure MPLS solution. The similar shape of the curves for $b = 25$ and



(a) Maximum Utilization (b) Flows

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

$b = 80$ 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 considerably. 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 an MPLS network with the use of OSPF

The D^{2}R-algorithm may as well be used to simplify a MPLS network with the help of OSPF. If f_i is equal to the total number of flows and b to the total number of links, the algorithm is operating like our pure 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. In case they match, one does not need to set up a MPLS path but can still use the existing OSPF path. While the average and maximum utilization remain equal to those of the MPLS approach the only difference consists in the number of MPLS pathes you have to set up. We therefore 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. There may be other reasons like QoS

Table 6
Number and percentage of changed flows

Networks	14		20		25	
	Def. OSPF	L.O. OSPF	Def. OSPF	L.O. OSPF	Def. OSPF	L.O. OSPF
Number of flows	47	20	119	122	164	150
Percentage of flows	25.8	11.0	31.3	32.1	27.3	25.0

to set up more MPLS pathes, however, from the performance point of view, in general less than one third of all 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 this paper we are able to offer valuable clues to decide about the preferred use of a local or a global TE system. 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 as illustrated in Figure 7. In principle, global TE offers

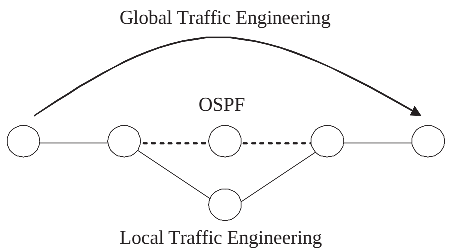


Figure 7. Local and global Traffic Engineering

Network	10	20	25
Changes	22	122	162
AHPC	-1.318	0.117	0.216
CLPP	1.818	2.454	2.574

Figure 8. Increase in the delay ($AHPC = 0$; $CLPP = 2$)

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. In a first approximation we therefore define the delay of a flow as the number of hops used. Figure 8 shows the increase in the average delay. While the first row states the total number of changes made the rows two and three describe the number of additional hops per change (AHPC) respectively the average number of links that the new path differs from the original path. On average we face no more than 2.6 changed links per path (CLPP). At 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 TE approach, we have to take a closer look at the path distribution of the chosen paths.

If the new path only bypasses the bottlenecks, we could easily develop a local TE approach. Otherwise, it will be complicated to construct a simple and local TE approach, as it will be very difficult to decide which particular flows to change and how to do so. That is, a local TE system would have to find an appropriate set of flows and subsequently signal the individual flows when and where to change their original path to avoid oscillations. However, there is no general way to do so. The number of hops in Table 7 describes

Table 7
Number of hops that the perfect path differs from the original path before/after the bottleneck.

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

the distance from the bottleneck that the optimized path differs from the original path. As shown in the Table 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 indicates the need for global TE, 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 the possibility of combining common network technologies like IGP and MPLS to a TE system. We introduced a fast linear optimization approach for MPLS networks, with an appropriate objective function for TE and compared default, optimized OSPF routing and MPLS TE to reveal the advantages of the latter. 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 of the network would be a fascinating but difficult task. Finally, 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*, RFC 2328, April 1998
8. Rosen, et al., *Multiprotocol Label Switching Architecture*, RFC 3031, Januar 2001
9. Awduche, et. al., *Overview and Principles of Internet TE*, RFC 3272, May 2002
10. Crawley, et. al., *A Framework for QoS-based Routing*, 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*, ZIB, Preprint SC-98-13, 1998
15. S. Köhler and A. Binzenhöfer *MPLS Traffic Engineering in OSPF Networks - A combined Approach*, TD304, University of Würzburg, 2003