Capacity Management: Using The Dual Solution of the Multi-Commodity Flow Problem to Set OSPF Weights - a Fast Heuristic

J. Murphy, X. V. Doan, R. Nelson† and R. Harris

Centre for Advanced Technology in Telecommunications
RMIT University, Melbourne
Victoria 3001 Australia
Contact Author: murphy@catt.rmit.edu.au

† ECSE Dept, Faculty of Engineering,
Monash University, Clayton
Victoria 3168 Australia

Abstract

We show that dramatic reductions in the maximum utilization of congested links in a network ("hot spots") can be achieved using the dual solution of the multi-commodity flow problem to set the link weights. Using a modern Linear Programming (LP) software package the time required to compute solutions is up to two orders of magnitude faster than schemes reliant on search techniques. Approximate solutions using split routing and single path routing are shown to produce near-optimal results. Single path solutions allow the use of MPLS over OSPF with a vastly reduced administrative overhead. We also show that sub-optimal (but vastly improved) solutions can be found so that MPLS need not be deployed at all.

1. INTRODUCTION

Many present-day routers use the OSPF or IS-IS protocol [1] for directing Internet traffic. These protocols route traffic on the shortest path between an origin and destination pair as defined by a suitable distance metric. Since this does not take account of the set of traffic demands, many situations occur in which the links on shortest path routes become congested while links on other paths remain relatively free [2]. For best effort traffic, this is not usually a problem. For traffic requiring guarantees on delay and jitter, however, congestion cannot be tolerated. The traffic needs to be "spread" so that as many links as possible are moderately loaded – some sort of constraint based routing is needed. One proposed solution is to use the traffic engineering capabilities of a new protocol such as Multi-Protocol Label Switching (MPLS) [2][3]. This is a powerful, but administratively more expensive solution, particularly if the explicit routing is invoked.

On the other hand, if the present routing protocol could be used to perform constraint based routing considerable savings in administrative overhead may be achieved. This is exactly what Fortz and Thorup proposed in their seminal work [4]. They showed constraint based routing of a similar quality to that of MPLS can be obtained just by "setting" the OSPF weights. To find this set of weights they use a sophisticated, computationally expensive search procedure.

In contrast, we proposed constraint based routing using an alternative approach [5] in which we used the dual solution of an LP formulation to determine the set of OSPF weights. The primal solution was then used to determine a set of splitting ratios. The prime advantage of this method was its potential to be very fast; the disadvantage was that splitting ratios were required, leading to implementation difficulties.

Each of these approaches focused on achieving a near optimal solution. As such, they were computationally intensive and required the changing of many OSPF weights. In contrast, Fortz and Thorup [6] recently published a paper in which they assert that any solution where many weight changes are made is not a *practical* solution for network managers. Obviously, as traffic demands change over time the weights have to change to keep the allocation of resources nearly optimal. In practice though, network managers are prepared to relax the optimality condition to minimize disruption to the network. This means network managers need a tool that allows them, firstly, to assess the effect that a weight change will have on a network before making the change; and secondly, to run an optimization program that effects the maximum change (towards a better solution) with the minimum number of weight changes.

Fortz and Thorup posed the question, "why are weight changes bad?"[6]. Their answer: weight changes have to be flooded throughout a network. The routers then have to recalculate new routes, which, if there are many weight changes, means that many flows have packets rerouted mid-flow. This in turn leads to out-of-order packets and subsequent degradation in TCP performance. The period of ensuing disruption could last for seconds and may take even longer before the network returns to equilibrium. Nevertheless, the true value of weight setting methods is that it is unnecessary to have a prescribed route for individual flows, with the attendant overhead that this involves.

Just as Fortz and Thorup [6] demonstrated a method to improve the routing of flows across the network with just a few weight changes, we show that by using the LP approach similar improvements in performance can also be achieved. The advantage of our method is that it is up to two orders of magnitude faster because of the use of advanced basis techniques and the incredible speeds available from modern LP solvers.

In this paper, we explore three solutions simplified from our original optimal solution [5] to allow easy implementation. The simplest of these only uses weight setting to move traffic off congested links in OSPF based networks. Nevertheless, the reduction in the maximum link utilization of a network with only a few weight changes is impressive and quite comparable to the results produced in [6]. The second uses only single-path routing and assumes a mechanism to choose a single path for flows when alternate shortest paths are available. The third is a hybrid solution of both equal-cost-multi-path (ECMP) and single paths. All three options are complementary to MPLS deployment, providing mechanisms that come close to optimal routing with much reduced administrative routing of individual LSPs (Label Switched Paths). The pure weight setting solution, however, does not require the use of MPLS at all

2. LP FORMULATION OF PROBLEM

We have formulated a particularly simple objective for our IP routing problem: we want to route as much traffic as possible down the cheapest route without overloading any of the links. This can be formulated as a multi-commodity flow problem. The commodity in this case is a flow of data traffic (a traffic demand) between an origin-destination pair. The cost of a link, (or more correctly, the cost per unit of flow) is set according to some predefined rule, and to prevent links becoming overloaded, each link has a "hard" capacity. We should note that "overloaded" means "greater than a given utilization for a particular link". For some links this may be close to 100%; for others with stringent delay requirements, it may be closer to 20%.

This problem is now described mathematically. For each commodity k, $\mathbf{P}^k$ denotes the collection of all directed paths from the origin node to the destination node, f(P) is the flow on one of these paths, and $\delta_{ij}(P)$ equals one if arc (i,j) is contained in the path P and is zero otherwise. This results in the following path-flow formulation of the LP problem.

$$\text{Minimize} \sum_{1 \le k \le K} \sum_{P \in \mathbf{P}^k} c^k(P) f(P) \tag{1}$$

subject to

$$\sum_{1 \le k \le K} \sum_{P \in \mathbf{P}^k} \delta_{ij}(P) f(P) \le u_{ij} \tag{2}$$

for all links (i,j) where u_{ij} is the capacity of each link and

$$\sum_{P \in \mathbf{P}^k} f(P) = d^k \tag{3}$$

where d^k is the size of the flow for each commodity, and k= 1…K and all $f(P) \ge 0$.

The size of the set of paths, $\mathbf{P}^k$, was actually set to ten, because few, if any, flows were ever found on paths with a path number greater than seven or eight. The constraints in (2) are known as the "bundle" constraints as they tie together all the flows by restricting the amount of flow on a particular link (i,j). If the capacities were sufficiently large, this problem would revert to K single commodity flow problems, the solution of which is to send every flow down its shortest path. This is precisely what happens in intra-domain routing using the OSPF protocol. If the network was lightly loaded, (ie capacities were sufficiently large) this routing method would be perfectly adequate. In fact, when a network is first dimensioned, one expects shortest path routing to provide good capacity management.

When the traffic demands begin to change and the network is no longer adequately dimensioned, areas of severe congestion ("hot spots") begin to occur. We must then solve the above LP formulation to find new feasible routes for the traffic demands. In the next section, we show that by using both forms of the LP solution, namely the "primal" and the "dual", a practical routing solution using the OSPF protocol can be implemented.

3. THE COMPLEMENTARY SLACKNESS CONDITIONS AND SHORTEST PATH ROUTING

The complementary slackness conditions have very interesting consequences for the routing problem. The path flow formulation contains a dual variable ω_{ij} for each link and another dual variable σ^k for each commodity k = 1…K. We define the reduced cost as

$$c_P^{\sigma,\omega} = c^k(P) + \sum_{(i,j)\in P} \omega_{ij} - \sigma^k$$

The path flow complementary slack conditions are stated in Ahuja et al [7] and are valid at optimality.

$$\omega_{ij}\,[\sum_{1\le k\le K} \sum_{P\in \mathbf{P}^k} \delta_{ij}(P)f(P) \; - u_{ij}] = 0 \quad \text{for all links (i,j)} \qquad (7)$$

$$C_P^{\sigma,\omega} \ge 0 \;\; \text{for all k = 1, K and all } P\in \mathbf{P}^k \qquad (8)$$

$$C_P^{\sigma,\omega} f(P) = 0 \;\; \text{for all k = 1,K and all } P\in \mathbf{P}^k \qquad (9)$$

It is further stated in [7] that eqns.(8) and (9) imply that:

σ^k *is the shortest path distance from origin node to destination node (commodity k) with respect to the modified costs* $c_{ij} + \omega_{ij}$ *and in the optimal solution every path from source node to destination node that carries a positive flow must be a shortest path with respect to the modified costs.*

3.1 Shortest Path Routing Using the Link Dual Prices

Routers make their routing decisions independently, based on shortest path calculations which are in turn based on a set of link weights (costs) for the whole network. From the results above we can see that if the modified costs, $c_{ij} + \omega_{ij}$, were used as the OSPF weights, then a traffic demand would automatically be routed on a path of cost σ^k. For demands where only one shortest path of cost σ^k exists, the actual routing solution is identical to the primal LP solution. Setting the OSPF weights to the modified costs will also initiate cases in which demands have multiple shortest paths of cost σ^k. For these cases the actual routing solution is more complicated and requires the calculation of node splitting ratios.

3.1.2 Finding the Node Splitting Ratios

Each router has a forwarding table in which each entry contains a destination and the link(s) a packet should take to reach that destination. If there is more than one shortest "modified cost" path available from a particular node to a particular destination one must assign a "splitting ratio" in the forwarding table that specifies the proportion of the incoming flow that should be sent on each outgoing link. Using the primal solution we find that for node "n" the splitting ratio for traffic destined for node "t" on link (n,j) is given by

$$R_{nj}^{nt} = \frac{\sum_{k\in D_t} \sum_{P\in \mathbf{P}_{sh}^k} \delta_{nj}(P)f(P)}{\sum_{l\in A_n} \sum_{k\in D_t} \sum_{P\in \mathbf{P}_{sh}^k} \delta_{nl}(P)f(P)} \qquad (10)$$

for (n,j) $\in A_n$, where A_n is the set of all outgoing links at node "n" and D_t is the set of flows going to t that pass through node "n". We note here that $\mathbf{P}_{sh}^{k}$ is the set of shortest "modified cost" paths that each flow can take to destination node "t" from node "n" and the general Boolean operator δ_{ij}(P) is equal to one if a link (i,j) lies on a path "P" and zero otherwise.

The splitting ratios are calculated and distributed to each node. In [5] we proposed a scheme in which the exact splitting ratios could be implemented. Unfortunately, there are no commercial routers that support this method thus we have proposed some routing schemes that can be implemented using present-day equipment. These are discussed in the next sections.

4. RESULTS

4.1 Three Approximate Splitting Ratio Cases

Topology	Total Demand	OSPF Max Utilization (Orig.)	OSPF Max. Utilization (Duals)	"1-0" Max. Utilization	Hybrid Utilization	No.of Splitting ratios
S	11,111	244%	131%	114%	104%	155
D	9,370	258%	120%	113%	107%	190
M	7,652	202%	112%	111%	107%	191

Table 1. Absolute Maximum Link Utilizations for with Link Metrics Inversely Proportional to Capacity

Legend: S – capacities very similar (400-600 units)
M - capacities quite different (200-800 units)
D – capacities much different (100-900 units)

The test network was a 56 node, 200 link (unidirectional) network consisting of two USA 28 node networks joined together. The total number of O-D pairs was 3080. For each O-D pair, ten paths were selected. The average capacity of each link was 500 units hence the total capacity of the network was 100,000 units. The cost of each link was set to the inverse of its capacity, which is the default metric recommended by Cisco. The set of link capacities were chosen randomly. We used the same topology with three different sets of capacities (S, D, M) that stand for "same", "different" and "much different". The upper and lower limits of the capacities of the link metrics are shown in the legend.

For each case (S, D, M) a demand matrix was generated multiple times. The total demand was chosen so that the LP problem would be feasible but the shortest path (routing problem) would produce overloaded links. The intention was to simulate networks that are in dire need of capacity management. In the (S) case the average total demand was 11,111 units and in the (D) and (M) case it was 9370 and 7652 units respectively (see Table 1). Thus in the LP solution the maximum utilization is 100%, and in a heavily loaded case we expect a number of links to have this utilization.

We then calculate the maximum link utilization when each set of traffic demands is routed down the shortest paths. For the (S) case this yields the figure 244%. Similar results hold true for the (D) and (M) case. Thus, we see that by using a routing scheme that implements the LP solution we completely improve the maximum utilization by up to 150%. This is reasonably obvious; what is important is how close our approximate routing schemes are to the exact case. We discuss the three cases below.

4.1.1 The “Duals” Case

As we pointed out in section 3, using dual prices and exact splitting ratios would yield an optimal solution that is unlikely to be implemented in practice. The first of the three approximations to the optimal solution, the “Duals” case is shown in Table 1. The OSPF weights are set to the modified costs, splitting ratios are ignored completely and the link utilizations are calculated using shortest path routing. This results in a reduction of the maximum utilization of between 90% and 138%. The maximum utilization for the optimal case is 100%.

4.1.2 The “1-0” Case

To more closely approach optimality, we can use an approximate set of splitting ratios that we label “1-0 splits”. A good proportion of the splitting ratios are already ‘1-0’. Splitting ratios that are not ”1-0” are termed “decimal” splits. It just remains to round up those that are greater than 0.5 to 1.0 and round down those that are less than 0.5 to 0.0 . Ties are broken randomly. We see that in the (S) case (Table 1) the maximum utilization is now reduced to 114%. This is a single path routing scheme and requires the “1-0” splits to be distributed to the routers. We see from Table 1 that the number of splitting ratios is small therefore this is not a prohibitive amount of state information.

4.1.3 The Hybrid Case

Finally we have a hybrid routing solution, a combination of ECMP and 1-0 routing. The decimal splits are deemed “50-50” if the split ratio is less than 0.75 or greater than 0.25. In this case the ECMP facility will split the flow in two. If the split ratio is greater than 0.75 or less than 0.25, the split ratio is deemed “1-0”. Again ties are broken randomly. For cases of “n” way splitting (n>2), of which none were observed here, a similar (but slightly more complicated) procedure is used. This hybrid solution yields a maximum utilization only a few per cent from optimal (see Table 1).

We found, in all our tests using inverse capacity as the link cost that the number of splitting ratios was in the order of 200 (see table 1). In contrast when we used unit costs there were a plethora of equal cost paths generated. This means that the number of splitting ratios is high to start off with and gets still higher (roughly 1000) thus the first two “cases” above are far less optimal. Fortz and Thorup [6] also only use inverse capacity as their starting cost as this was deemed to be the best performer.

4.2 A (Re)-Routing Method

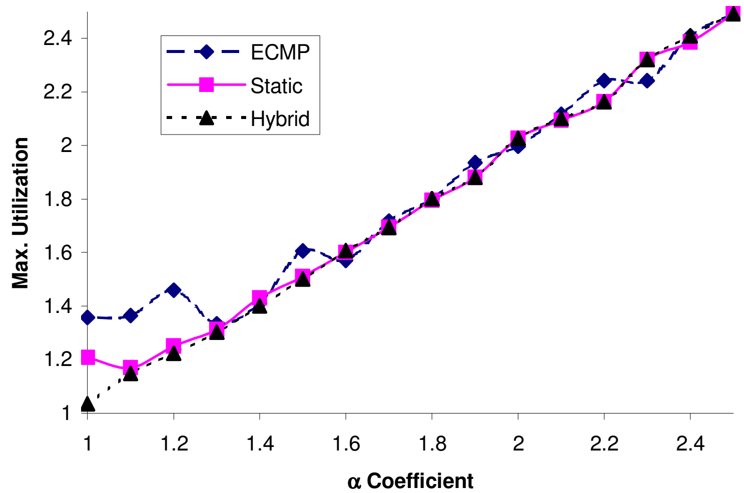

Figure 1. Max Utilization of the three cases vs α, the capacity scaling coefficient for the 56 Node Graph (S)

We will now show (as in [6]) that it is possible to route a significant proportion of a given set of traffic demands with only a few weight changes. We take an (S) topology network with a set of demands totalling 11,193 units of flow. In Figure 1 the link capacities are scaled by a coefficient α and the network is optimised for each separate value of α. The starting value α_s is just above the maximum utilization found when this set of demands is routed down the shortest paths. This means all flows use shortest paths, no links are full and there are no dual prices. Our aim is to successively shift the flows off the congested links by lowering the "effective capacity" of the links.

For each value of α we solve our LP formulation thus in the actual network the best possible maximum utilization would be α. Reference to Figure 1 shows that our approximate solutions have similar maximum utilizations until $\alpha = 1.3$. For $\alpha < 1.3$, the "duals" case actually gets worse and for $\alpha < 1.1$ the "1-0" case gets slightly worse. The hybrid case is nearly optimal for all values of α. For $\alpha = 1$ the maximum utilizations of the three cases are almost the same as those found in Table 1 for the (S) (ave.) case.

α Coeff.	No.of Duals	Total No. Split Ratio	Timing (s)
2.6	0	0	0.18
2.4	1	1	0.23
2.2	1	1	0.14
2	1	1	0.151
1.8	1	1	0.151
1.6	2	31	0.17
1.4	4	17	0.541
1.2	10	58	1.252
1	24	166	7.1

Table 2 The capacities are scaled by a coefficient α and the network is optimised. 56 node (S) case. 11,193 units of flow.

If we now examine Table 2 we see that the number of dual prices is small until α approaches 1. Moreover from Figure 1 we have seen that the "duals" case is a good approximation to the optimal solution for α >1.3. Therefore we conclude that we can effect large changes in maximum utilization by using only a few dual prices to reset the OSPF weights. This is entirely consistent with the reported findings of [6]. The difference in our case is that our calculation time is much faster.

4.2.1 Calculation Times

The strength of the above method as a routing or capacity management tool is that the parts that take the most time only need to be carried out once while the parts that need to be repeated multiple times take little time at all. For each calculation these steps need to carried out:

(1) Generating the paths for each flow for a given topology.
(2) Reading and writing of an input file.
(3) Running the LP program until the desired level of accuracy (α).
(4) Calculating the splitting ratios.
(5) Running the shortest path calculation for the three approximations to calculate the link utilizations

We will consider two cases, one of 56 nodes and one of 90 nodes (300 links). In either case steps (1) and (2) need only be carried out once thus our calculation time for any capacity management "tool" we may envisage is the sum of steps (3), (4) (5).

For 56 nodes (S), step (1) took three minutes, step (2) took 20 s, step 3 took between 0.4 and 10 s (see Table 2), step (4) took 6 s and step (5) took 0.3 s each. We should note for the "duals" approximation splitting ratios need not be calculated.

For 90 nodes (S), step (1) took 30 minutes, step (2) took 1 m, step 3 took between 0.4 and 30 s, step (4) took 15 s and step (5) took 1 s each. The resultant improvement in maximum utilization was similar to the 56 node case.

By Examining Table 2 we see that the maximum utilization for the 56 node network can be halved in 1-2 seconds using the "duals" case, steps (3) and (5). Only two dual prices are needed. We performed the same calculation for the 90 node network and halved the maximum utilization in only six seconds. This is two orders of magnitude faster than Fortz and Thorup's calculation for a similar size network [6] (using a 1.7 GHz Pentium 4). We believe this has important implications for the construction of a much improved network management tool along the lines of that described in [8].

4.3 "Hot Spot" Remover

This method works well because even heavily loaded networks often only have a small fraction of their links congested (though some heavily so) thus few dual prices and few splitting ratios are generated. Those highly congested links are known as "hot spots" and are caused by changes in the original demand matrix.

It is precisely these 'hot spots" or "bottlenecks" that cause the QoS to degrade rapidly in a network. So we see that, as in [6], our primary purpose is not to achieve optimality (though our approximations can be good) but to remove "hot spots" with as few weight changes as possible. It is here that our method works particularly well. It is a natural solution for this problem. Each iteration of the dual Simplex method (which with CPLEX 7.1 [9] is very fast) will detect the most congested link, remove the largest flow and find a feasible path for it. In addition, the new path is a shortest modified cost path hence we can input these new link costs directly as the OSPF weights.

By examining Table 2, we can see a poignant example of the power of this method. Only one dual price is needed to lower the maximum utilization from 2.6 to 1.8. Furthermore, there is only one splitting ratio and four alternate paths (this is gleaned from the solution file). It turns out, however, that 81 flows are rerouted. 81 misrouted flows out of 3080 is a tiny fraction and would cause little problem if spread across the network, but 81 flows routed on a heavily loaded link would cause a severe bottleneck.

5. DISCUSSION – THE ADVANTAGE OF SPEED

One great advantage of this technique for calculating OSPF weights over that presented by Fortz and Thorup [6] is the speed of calculation. The LP calculation presented here takes only a few tens of seconds, as opposed to tens of minutes. Since the aim of these techniques is to provide a tool for network operators to investigate changes to the routing parameters, this speed is not directly critical to the operation of the network. Nevertheless, it does provide significant improvements. The first of these is the ability to be interactive. Network operators who wish to calculate new weights with this technique could do so, view the effects and then choose to calculate alternative sets in an interactive fashion. The second advantage is in performing "what-if" analysis with a number of variable parameters, eg a new range of predicted demands. This tool could be used to calculate weights for dozens or even hundreds of scenarios in a reasonable time; minutes to hours rather than days to weeks.

This technique is sufficiently fast that it would be possible to incorporate it in an online mechanism to react to network changes. This, however, is not likely to be desirable in the normal situation due to the adverse effects of the routing changes caused by changing OSPF weights. Still, if a tool is being used to calculate the link weights, it could be useful to have answers available quickly in a network emergency caused by a truly unanticipated or especially dramatic change in network conditions eg the formation of a sudden hot-spot.

6. IMPLEMENTATION

In our previous paper [5], we discussed the implementation of a traffic management system (ie managing a changing set of traffic demands) that needed splitting ratios. It turns out that this was not likely to be implemented. In this paper we have proposed three solutions (cases) using approximations to splitting ratios. We believe all three solutions can be practically implemented. We assume here that a practical size OSPF network is approximately 50 nodes and that the timing of the implementation of the routing decisions is a human factor.

6.1 The “Duals” Case

This technique provides “hot spot removal” (as in [6]) for standard OSPF networks and so has clear application to the management of common backbone networks. All that is required is that the new weights (the modified costs) be distributed to all the routers. Statistics for the traffic matrix would be gathered using Netflow [10] or Service Level Agreements (SLA).

It is also possible to run MPLS and route the LSPs using OSPF. Using the Interior Gateway Protocol (IGP) for routing LSPs is common in many MPLS deployments. This has the advantage that complex explicit routing protocols are no longer needed and traffic matrix statistics are much easier to gather.

6.2 The “1-0” Case

The “1-0” splitting option is not easily applicable to pure OSPF-based IP routing. The best application for the “1-0” routing case is using MPLS. Although LSPs can be all administratively routed, it is simpler to allow the majority of LSPs to route following the IGP and only administratively route a few to obtain some routing improvement. The 1-0 case supports this situation directly because it provides weights that go a long way towards optimising the routing of the majority of LSPs that follow the IGP. Those flows that have alternative paths can be administratively confined to one path using loose source routing [11]. This has the potential to provide nearly optimal routing with minimal administrative state information requirements.

6.3 The Hybrid Case

This case is closest to optimal routing. This could also be implemented using MPLS in two different ways. The first would be the same as the “1-0” case with the addition that, where ECMP splitting is required, multiple LSPs would have to be defined, one source-routed to each of the available paths. Then the traffic could be split between these paths at the entrance to the network using an IGP. The second technique would be not to use MPLS at all for the majority of traffic, but to route it using the IGP following the paths calculated with the (nearly) optimal weights. MPLS would be reserved for those flows on paths that required 1-0 splitting. This technique has similarities to those networks where MPLS is reserved for traffic that has a particular status separate from the bulk of best effort traffic, such as VPN traffic or higher QoS class traffic.

CONCLUSION

Previous work formulated the routing problem as a multi-commodity flow problem and showed that, when the dual solution values are assigned to the OSPF metrics and the primal solution is used to calculate node-splitting ratios, constraint based routing can be performed using the OSPF protocol. This paper extends this work by showing that simplified versions of this mechanism can provide nearly optimal routing with direct practical application to commonly deployed routing protocols. The techniques presented here have significant speed advantages over similar techniques proposed elsewhere.

ACKNOWLEDGMENTS

The authors would like to thank the generous support of the ATcrc in this project. John Murphy would like to thank Leith Campbell, Bill Lloyd-Smith, Greg Daley, Alex Kist and Brenton Steele for helpful discussions. He would also like to thank Andrew Cassin for his untiring efforts in supporting the computational work.

BIBLIOGRAPHY

[1] C. Huitema, *Routing in the Internet* (Prentice Hall 1995)

[2] X. Xiao, A. Hannan, B. Bailey and L.M. Ni, *Traffic Engineering with MPLS in the Internet,* IEEE Network **14** Mar/Apr (2000)

[3] X. Xiao and L.M. Ni, *Internet QOS: A Big Picture,* IEEE Network **13** Mar/April (1999)

[4] B. Fortz and M. Thorrup, *Internet Traffic Engineering by Optimizing OSPF Weights*, 19th Annual Conference of IEEE Computer and Communications Societies Proceedings **2** p 519 INFOCOM 2000

[5] J. Murphy, R. Harris and R. Nelson, *Traffic Engineering Using OSPF Weights and Splitting Ratios*, IFIP Interworking 2002

[6] B. Fortz and M. Thorrup, *Optimizing OSPF/ISIS Weights in a Changing World* JSAC **20** May (2002)

[7] R. K. Ahuja, T. L. Magnanti and J. B. Orlin, *Network Flows* (Prentice Hall 1993)

[8] *NetScope: Traffic Engineering For IP Networks,* A. Feldmann, A. Greenberg, C. Lund, N. Reingold and J. Rexford, IEEE Network **14** Mar/Apr (2000)

[9] CPLEX 7.1 ILOG, Inc. Mountain View, CA

[10] Http://www.cisco.com/warp/public/732/Tech/netflow/

[11] E. Rosen, A. Viswanathan and R. Callon. *RFC 3031 Multiprotocol Label Switching Architecture,* (January 2001). http://www.ietf.org/rfc/rfc3031.txt