Engineering the Multi-Service Internet: MPLS and IP-based Techniques

P. Trimintzios[α], L. Georgiadis[β], G. Pavlou[α], D. Griffin[γ], C.F. Cavalcanti[α], P. Georgatsos[δ] and C. Jacquenet[ε]
[α] C.C.S.R./University of Surrey, UK
[β] *Aristotle University of Thessaloniki, Greece*
[γ] *University College London, UK*
[δ] *Algonet S.A., Greece*
[ε] *France Telecom, France*

ABSTRACT

IP Differentiated Services (DiffServ) is seen as the framework to support quality of service (QoS) in the Internet in a scalable fashion, turning it to a global multiservice network. In this context, integrated service/network management and traffic control mechanisms are of paramount importance for service provisioning and network operation, aiming to satisfy the QoS requirements of contacted services while optimising the use of underlying network resources. In this paper, after briefly introducing an architectural framework for integrated service/network management and control, we concentrate in its traffic engineering aspects comparing and contrasting two different approaches: MPLS-based explicit routed paths and IP-based hop-by-hop routing. We consider relatively long-term network dimensioning based on the requirements of contracted services and subsequent dynamic route and resource management that react in shorter time scales to statistical traffic fluctuations and varying network conditions.

I. INTRODUCTION

TEQUILA (Traffic Engineering for QUality of service in the Internet at LArge scale) is a European collaborative research project looking at an integrated architecture and associated techniques for providing end-to-end QoS in a DiffServ-based Internet. In TEQUILA we have produced a framework for Service Level Specifications (SLSs) [God01], we have designed an integrated management and control architecture [Trim01] and we are currently investigating both MPLS- and IP-based techniques for traffic engineering. In this paper we present, techniques for network dimensioning, dynamic route and dynamic resource management, contrasting MPLS and IP-based approaches.

The rest of this paper has the following structure. In section II we present a functional architecture for supporting quality of service in IP differentiated services; presenting briefly all its aspects but concentrating on the architectural decomposition of the traffic engineering part. In section III we present techniques for network dimensioning, in section IV techniques for dynamic route management and in section V techniques for dynamic resource management. We finally conclude with a brief summary in section VI.

II. A FUNCTIONAL MODEL FOR QOS

In order to support end-to-end QoS based on Service Level Subscriptions (SLSs), within the TEQUILA project we have defined a functional architecture [Trim01] whose main components are depicted in Figure 1.

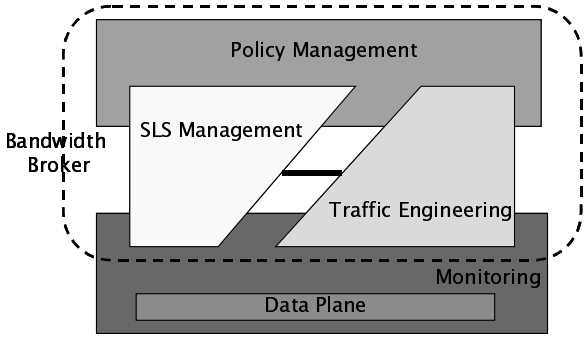


Figure 1: A functional model for providing QoS

This architecture includes both control and data plane as well as management plane functionality. The management plane aspects can be seen as a detailed decomposition of the concept of Bandwidth Broker (BB) [Nich99]. In our architecture, the BB is realized as a hierarchical, logically and physically distributed system. Every Autonomous System (AS) should deploy its own BB, while the end-to-end QoS requirements supported by the collaboration of several such BBs over the ASs involved in the forwarding path.

The SLS Management part of the architecture provides the interface to customers for service subscriptions and subsequent invocations; it should be noted that a service offering may comprise more than one SLS. The SLS parameters and their semantics are specified in [Gode01]. The data plane functionality includes the DiffServ Per Hop Behavior (PHB) implementation [Blake98] and possibly additional explicit path support functionality such as Multi-Protocol Label Switching (MPLS) [Ros01]. Monitoring is one of the most important parts of this architecture since its services are required by almost all the other components. The Policy Management part of the architecture allows administrators to enforce policies on both the SLS and Traffic Engineering parts. In order to meet the subscribed SLS

requirements while utilizing network resources efficiently and reliably [Awd00], each AS, including its BB, must be carefully engineered.

Traffic Engineering Components

We will pay special attention to the Traffic Engineering (TE) subsystem of the functional model of Figure 1, which is further decomposed into the modules shown in Figure 2.

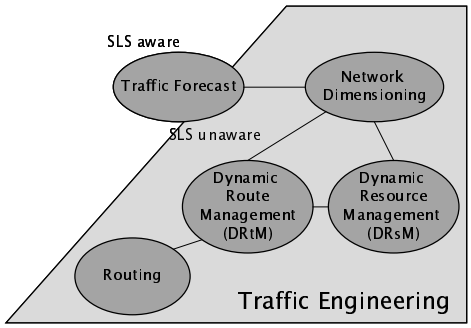


Figure 2: Traffic engineering modules

Traffic Forecasting (TF) is mostly part of the SLS Management subsystem, and it provides aggregate traffic predictions to the rest of the system, utilizing information from the subscribed SLSs as well as measurements and historical data [Srid01]. The SLS aware part of TF is part of SLS Management while the SLS unaware part is part of Traffic Engineering. The produced traffic matrix contains information about ingress-egress bandwidth, delay and loss requirements. Having this information, the traffic engineering task is decomposed in two levels corresponding to the time- and state- dependent TE described in [Awd00]. The higher level intends to provide long-term guidelines for sharing the network resources and is implemented by Network Dimensioning (ND). The lower level intends to manage the resources allocated by Network Dimensioning during the on-line system operation in order to react to statistical traffic fluctuations and special network conditions and is implemented by Dynamic Route and Resource Management (DRtM/DRsM). DRtM manages the routes and route bandwidth defined by ND. Similarly, DRsM manages the packet queuing and forwarding resources at each network node according to the guidelines provided by ND.

In the following we provide our approach to traffic engineering under two assumptions on network capabilities: networks that are MPLS capable and networks that implement classic shortest-path based routing with the recent QoS extensions [Apo98].

III. NETWORK DIMENSIONING

A. MPLS-based Approach

The MPLS approach to Network Dimensioning utilizes the set-up of explicitly routed paths without bandwidth reservation. This is done in order to provide guidelines to DRtM and DRsM on how to best accommodate the predicted traffic.

The entries of the traffic matrix are the traffic trunks [Li98]. Each trunk is the aggregation of a set of traffic flows characterized by the same ingress and egress nodes and performance requirements. Aggregating flows into trunks results in fewer entries thus increased scalability [Awd99]. In the multi-class setting we use in this work, the traffic class (called QoS-class in the remainder of the paper) of the trunk is defined by the Ordered Aggregate (OA) [Blak98] bandwidth, maximum delay and loss probability requirements.

A traffic trunk follows the pipe model, i.e. each traffic trunk is associated with one ingress, one egress node. In this work we enhance the traffic trunk model to cater for the hose model which is associated with one ingress and more than one egress nodes [God01]. More specifically, the bandwidth (traffic rate) that a hose trunk requires at the ingress node can be directed to any of the trunk egress nodes. This has implications as to the efficient bandwidth allocation within the network as illustrated in Figure 3.

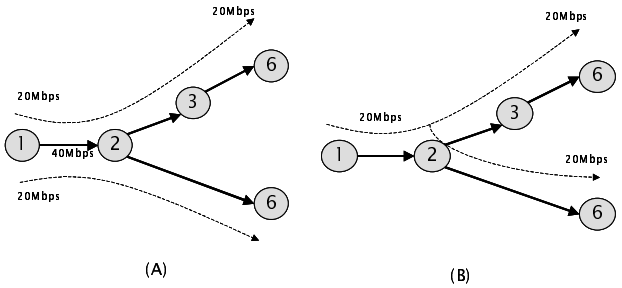


Figure 3: Efficiency implications of the hose trunk model.

In (A), in order to serve the hose trunk's requirements we define 2 paths, and allocate bandwidth of 20Mbps to each of these paths. Hence with this approach we just allocate bandwidth of 40Mbps on link (1,2). However, since at most 20Mbps can enter from node 1 (although a fraction of it may be transferred to egress nodes 4 and/or 5), it is clear that reserving bandwidth of 20Mbps on link (1,2) suffices. To effect this bandwidth saving, in (B) we define a tree and allocate bandwidth of 20Mbps to each branch of it. Consequently, in this work instead of searching for best paths to satisfy our objectives we consider trees and associate each brunch of the tree with a certain bandwidth, the "tree bandwidth". Trees are then decomposed in a number of Label Switched Paths (LSPs). However, we do not directly associate bandwidth with an LSP. Instead, the capacity assigned to a PHB on a given link is the sum of the bandwidth requirements of the trees passing through that link. Note that this does not require changes in the LSP path set-up mechanism.

Objectives

The primary objective of network dimensioning is:

I. Satisfy the QoS-class requirements of all trunks as long as their traffic is within the trunk's bandwidth limit.

This objective provides a feasible solution that satisfies the trunks requirements. However the design objectives can be further refined to incorporate other traffic engineering related requirements. Those are:

II. Avoid overloading parts of the network while other parts are underloaded.

This results in accommodating better unpredictable (e.g. best-effort) traffic while failures disrupt smaller amount of traffic.

Minimize the overall network cost.

With each link l and a given OA, we associate a cost function $f(x)$, where x is the bandwidth allocated to the OA. This cost function may represent the link utilization but it may also be a function determined by administrative policies. We assume that *$f(x)$ is convex*. Objective II above can be associated with the following optimization criteria:

$$\min \quad (\max_{l \in E} F_l) \tag{1}$$

$$\min \quad \sum_{l \in E} F_l \tag{2}$$

The first criterion can be further refined to a lexicographic optimization problem [Geo01], where the optimal solution is not determined only by the "worst" loaded link but from the whole vector of link loads. The second criterion attempts to maintain a low overall network cost. It is possible to define a compromise between the two criteria as follows:

$$\min \quad \sum_{l \in E} (F_l)^n , \; n \geq 1 \tag{3}$$

When $n = 1$ the formula is reduced to (2), when $n = \infty$ to (1).

The above optimization problem has as constraints the end-to-end delay and loss requirements of each trunk. It turns out that incorporating these constraints into the optimization problem one can use gradient projection algorithms [Ber92] to solve the optimization problem in (3). At each iteration of the algorithm, minimum weight paths or trees (depending on the traffic model) are sought. Moreover, additional additive constraints on the paths (trees) must be considered due to the end-to-end QoS constrains. The problem of finding routes satisfying these constraints is NP-complete. Given that this is only a part of the problem we are addressing, we can make a simplification and transform these constraints to a number of hop constraints. This can be done by assuming that we have a worst-case delay bound for each PHB on every link as well as a bound on the loss probability (note that these bounds are relatively easy to obtain for certain schedulers). By considering the end-to-end delay and packet loss probability as the sum of the per-link per-PHB and packet loss probabilities, it is possible to translate this end-to-end constraint into a bound on the path (tree) hop-count. As a result of this simplification, the minimum cost path (under the hop-count constraint) algorithm becomes of polynomial complexity. However, for the host traffic trunk model, one has to implement a minimum weight tree algorithm; this problem is well known to be NP complete and hence we must rely on heuristics. In any case, the choice of translating the end-to-end QoS requirements into hop-count constraints still simplifies the heuristics that are to be employed. Note that since ND provides directives within which DRtM and DRsM should operate, an exact optimization is not critical at this point.

An additional issue arises by the need to define paths or trees for each of the defined QoS classes. There are two alternative approaches to handle this problem. One is to optimize over all the QoS-classes at once. The other alternative is to solve a series of optimization problems by staring from the one which has the greatest priority, and reducing the resources consumed by this QoS-class. The QoS-class priority is a policy-based decision.

As a result of the solution to the optimization problem, a number of trees with associated tree bandwidths are determined for each ingress node and each QoS class. These trees are downloaded to the DRtMs responsible for the given ingress node. In addition, the bandwidth of each link PHB that is required to carry the tree traffic is calculated and downloaded to the corresponding DRsMs. In addition ND may specify the minimum and maximum values by which the actual bandwidth allocated to a PHB by DRsM during the on-line operation, may deviate from its nominal required value.

B. IP-based Approach

The IP-based traffic engineering approach attempts to accommodate the traffic requirements of the traffic trunks entering the network by appropriately specifying the operational parameters of the standard IP intra-domain routing protocol, namely OSPF [Moy98]. The operational parameters refer mainly to link costs and hashing mechanism based on which the OSPF shortest path routes are determined. Hence, in the IP-based traffic engineering approach,

- Link weights determine the traffic routes for the various traffic trunks.
- The routes and the traffic load of each of the traffic trunks determines the link loads
- The link loads and the cost functions associated with each link load determine the *system cost* associated with the particular choice of link weights

The optimization problem can then be formulated as follows.

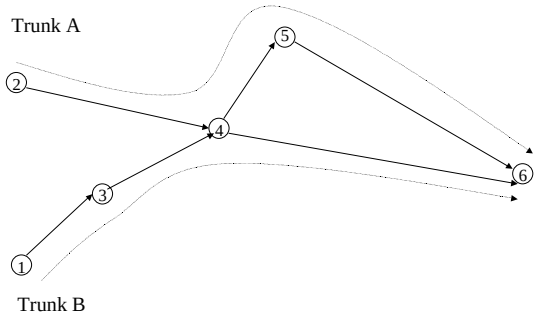
Determine the link weighs so that the overall system cost is minimized

At the outset, the constraint of having to specify the routes based on shortest paths imposes restrictions of the route design that are not present in the MPLS approach. Therefore, one expects that in general the MPLS-based optimization can achieve smaller system cost than the IP-based approach. However, in [Wan99] it was shown that the OSPF weights can be determined so that the resulting system cost is the same as the one that would be achieved by the MPLS approach. The algorithm in [Wan99] requires that the routers employ Equal Cost Multi Path (ECMP), i.e. each router performs load balancing on routes that have equal cost to a given destination. The parameters for load balancing are defined based on the bandwidth associated with each route through the solution of the optimization problem.

There are two obstacles to the above-mentioned approach to IP-traffic engineering.

First, it is required that the ECMP load balancing is performed based on the route bandwidths determined by the solution to the optimization problem. Some type of weighted round robin schedulers can achieve this requirement fairly easily, if packets are allowed to arrive out of order to the destination. However, if packets-in-order is a requirement, then some kind of hash function on flow identification has to be performed [Tha00][Hop00]. Therefore the hash function has to be designed based on the determined route bandwidths. This requires either a priori knowledge of related statistics, or the development of sophisticated hash functions. In addition, is should be ensured that the hash mechanisms employed at each router are consistent. In fact, placing the restriction that load balancing be made by splitting the load equally renders the OSPF approach sub-optimal [For00].

Second, even assuming the ECMP capability and the availability of appropriate hash functions, the inclusion of QoS constraints other than bandwidth in the above formulation places strict constraints on the IP-based approach. Consider for example placing hop-constraints on the traffic trunk routes. In the example in Figure 4, if the links have capacity 1, the only possible solution for the traffic load brought by trunks A and B is the one shown. Of course, this can be achieved with the IP-approach by defining appropriately the link costs so that the costs of paths (4, 5, 6) and (4, 6) are the same, and by routing explicitly Trunks A and B on the paths shown. However, this is in effect the MPLS approach. In the general case, specifying routes in this manner will cause more overhead than the MPLS approach, since routing will be based on flow IDs rather than label switching.



•Trunk A: max hop-count 3, bandwidth 1
•Trunk B: max hop-count 3, bandwidth 1
•All link capacities are equal to 1

Figure 4: An example of traffic trunk routing.

While the discussion above shows that the IP-based approach may require complicated ECMP load balancing, it has the advantage that it is readily implementable based on the widely available OSPF protocol and it scales better than MPLS. Moreover, studies have shown that in certain networks the performance of the IP-based approach with simple ECMP load balancing is not far from the MPLS approach [For00]. If the proposed QoS related extensions to OSPF are implemented [Apo98], then some of the above-mentioned issues may be resolved.

In the model we consider in this work, we have to also take into account the traffic trunk hose model. As discussed above, bandwidth efficiency can be achieved in such a model if with each traffic hose there is at least one associated tree containing all the egress nodes of the hose, and having as source the hose ingress node. With the IP-approach, the tree associated with the defined traffic hoses can be naturally defined as follows. Once the link weights have been defined, the shortest path tree, S_i, from an ingress node i to all egress nodes can be defined. For each of the traffic hoses entering the network from the given ingress node, the associated tree is the sub-tree of S_i that contains all the egress nodes of the hose.

Having defined the hose trees, the links loads can now be determined and the system cost function can be calculated. A heuristic using some of the ideas in [For00] is then used to modify the link costs in such a manner that the systems cost is improved. The heuristic is in effect a local search technique, whereby the links whose weights are modified are these which emanate from the same node as the link with the largest cost function.

The algorithm is applied successively for each of the PHBs defined in the system. Hence, for each of the defined PHBs, different weights are assigned on each link. These weights are downloaded to the routers and are used to populate the forwarding tables, one for each of the defined PHBs. In addition, the algorithm provides the link bandwidth allocated to each of the PHBs. This information is downloaded to Dynamic Resource Management which configures the routers.

IV. DYNAMIC ROUTE MANAGEMENT

A. MPLS-based Approach

In the MPLS approach, the Dynamic Route Management (DRtM) component is a distributed component located at the edge routers, responsible for managing the routing processes in the network according to the guidelines provided by Network Dimensioning. This amounts to:

- Setting up traffic forwarding parameters at the ingress node, so that incoming traffic is routed to LSPs according to the bandwidth determined by Network Dimensioning.
- Modifying the routing of traffic according to feedback received from Network Monitoring
- Issuing alarms/warnings to Network Dimensioning in case available capacity cannot be found to accommodate new connection requests

During initialization, Network Dimensioning provides DRtM the set of (hose) traffic trunks, $\mathbf{T}$ which are to be managed with DRtM. The common characteristic of this set of traffic trunks is that they all have as ingress node the node for which the given DRtM is responsible. With each traffic trunk

$T \in \mathbf{T}$ the following information is provided:

- The set of trees S_T to which traffic belonging to T is to be routed, as well as the bandwidth of each of these trees (the bandwidth of a tree is the bandwidth allocated to each of the links of the tree).
- The PHB treatment of traffic belonging to T
- The end-to-end delay and loss probability (upper bound) of traffic belonging to T

DRtM also requests from Network Monitoring statistics about the load incurred by various groups of "addresses". This statistical information is used by DRtM to allocate address groups to each of the traffic trunk trees, according to the bandwidth assigned to these trees. Based on this allocation, the LSP forwarding table at the ingress router is populated.

During system operation Network Monitoring informs DRtM about the QoS performance (end-to-end delay, loss probability and used bandwidth) of the traffic routed through the LSPs managed by DRtM. In addition, Network Monitoring informs DRtM about the QoS performance of the network PHBs used by the managed LSPs.

The monitoring of PHB QoS performance is used by DRtM to take proactive measures. Specifically, DRtM may avoid routing traffic to LSPs using the PHBs whose QoS performance in terms of delay and loss probability becomes critical, even though end-to-end performance deterioration on these LSPs may not have been observed. Hence actions at this stage attempt to avoid the deterioration of end-to-end QoS metrics and in addition help in relieving the load on the congested PHB.

The monitoring of LSP QoS performance is used by DRtM to take reactive measures. Specifically, DRtM will avoid traffic routing on LSPs whose QoS performance is already critical. However, some end-to-end QoS performance deterioration may have occurred at this point.

Based on the information received by Network Monitoring, DRtM may reassign some of the address groups to the various managed trees and hence update the LSP forwarding table at the ingress router. During this process, mechanisms are employed to ensure that during reassignment the packets-in-order condition is satisfied. If appropriate LSPs for the reassignment cannot be found, DRtM issues alarms to Network Dimensioning, which in turn may take more global actions in order to relieve the congestion.

B. IP-based Approach

The DRtM component in the IP-based TE approach is centralized and much closer tied-up to ND. It main objective is to update link weights during the on-line system operation in order to adjust to traffic fluctuations. Since a small change in the weight of a link may lead to a large number of route changes and hence to a large amount of load shifting at various parts of the network, it is required that DRtM has a global network view. This is the reason we chose to implement DRtM as a centralized component.

The deployment of load-sensitive change of the link metrics is hampered by the overhead imposed by the link-state update propagation, leading to significant route flapping, since paths are selected based on “out-of-date” information. In some cases it is possible to overcome this problem, by updating the costs only for long-lived flows [Shai99]. Though this approach works well, it has the drawback that it is very difficult to draw the line between long- and short-lived flows; in addition, it requires the use of pre-computed paths (e.g. LSPs) for the short-lived flows. Our main research concern on this issue of dynamically adjusting the link costs is on the definition of heuristics that, based on thresholds on link loads, drive to route adjustments without causing excessive route flapping. This work is still ongoing and it is mainly based on experimentation.

V. DYNAMIC RESOURCE MANAGEMENT

One of the requirements of QoS provisioning is a means for logically or physically partitioning network resources so that different traffic types do not interfere to the extent that they degrade the performance of each other. Resource partitioning, on the other hand, may mean that the network is inefficiently utilized if the width of the allocated partitions is not in accordance with the requirements of the actual load. In this case, resources allocated to one traffic type may exceed demand while insufficient resources are available for another traffic type where the allocated resources have been underestimated. This may result in higher than expected blocking or dropping rates for the other traffic types, which impacts their performance, and hence the delivered QoS. For this reason it is desirable to dynamically manage resource partitioning.

Dynamic Resource Management (DRsM) has distributed functionality, with an instance attached to each router. In both MPLS and IP TE approaches, it aims at ensuring that link capacity is appropriately distributed between the PHBs sharing a link by appropriately setting buffer and scheduling parameters according to ND directives, constraints and rules. Specifically, DRsM receives estimates of required resources for each PHB in terms of minimum and maximum bandwidth to be allocated to that PHB, a minimum bandwidth to be allocated in time of congestion (competition from the other PHBs) together with the maximum delay and packet drop probability to be experienced by packets using that PHB. Through these parameters ND specifies an acceptable operational range for the PHB's bandwidth, which has been calculated, based on the traffic forecasts it has received from the SLS Management system. Within the bounds of this margin, DRsM is free to dynamically manage resource reservations (i.e., the effective resources required to cope with unexpected SLS invocations, for example). Compared to ND, DRsM operates on a relatively short time-scale (order of minutes).

DRsM triggers ND when network/traffic conditions are such that its algorithms are no longer able to operate effectively, e.g. due to excessive high priority traffic, link partitioning is causing lower priority/best effort traffic to be throttled. DRsM may issue over- or under-load alarms to ND

respectively if the higher margin is closely approached, or if the PHB's rate has been below the lower margin for a predetermined time.

In its simplest form the DRsM is responsible for tracking the utilization of a PHB through the services of a Monitoring system, which is capable of issuing alarms when defined thresholds on PHB rate have been crossed. When lower thresholds are crossed, Monitoring triggers DRsM and the PHB is considered to be under-utilized. DRsM should reduce the allocated bandwidth to allow other PHBs to be allocated additional link resources should they require them. If the PHB is overloaded and the upper threshold has been crossed then the bandwidth should be increased if sufficient link capacity is available.

While this illustrates the role of DRsM in managing a single PHB/queue, the complete task of DRsM is to manage the resources of all the PHBs defined on a link by distributing bandwidth and buffer space among them. DRsM distributes spare link capacity between the PHBs when the sum of the demands is less than the link capacity. When the sum of the demands is greater than the link capacity DRsM will allocate the minimum congestion bandwidth to each PHB and distribute the remaining link capacity in proportion to the demands of each PHB.

VI. SUMMARY

In this paper, we presented first an architectural model for supporting QoS in a differentiated services Internet. We then focused in traffic engineering aspects, considering network dimensioning, dynamic route and dynamic resource management and presenting both MPLS and IP-based techniques. A key aspect in our approach is that network dimensioning is based on a traffic matrix produced through the contracted SLSs, as well as measurements and historical data. SLSs provide the targets to satisfy and constitute the "raison d' etre" for the proposed functional model and the associated traffic engineering mechanisms. The issue of MPLS vs. IP-based traffic engineering is becoming an important topic, attracting the attention of the networking community. We plan to come back to the subject in future papers, presenting our techniques and algorithms in more detail, supported by quantitative results.

ACKNOWLEDGEMENTS

This work was undertaken in the Information Society Technologies (IST) TEQUILA project, which is partially funded by the Commission of the European Union.

REFERENCES

[Apo98] G. Apostolopoulos et al. "QoS Routing Mechanisms and OSPF Extensions", RFC 2676, Experimental, August 1999

[Awd99] D. Awduche et al. "Requirements for Traffic Engineering over MLPS", RFC 2702, Informational, September 1999

[Awd00] D. Awduche et al. "A Framework for Internet Traffic Engineering", Internet Draft <draft-ietf-tewg-framework-02.txt>, work in progress, July 2000

[Ber92] D. Bertsekas and R. Gallager, Data Networks, Prentice Hall, 1992

[Blak98] S. Blake et al. "An Architecture for Differentiated Services", RFC 2475, Informational, December 1998.

[For00] B. Fortz and M. Thorup, "Internet Traffic Engineering by Optimizing OSPF Weights", in Proc. IEEE INFOCOM, Israel, March 2000

[Geo01] L. Georgiadis et al. "Lexicographically Optimal Balanced Networks", to appear in Proc. IEEE INFOCOM, Alaska, April 2001

[God01] D. Goderis et al. "Service Level Specification Semantics and Parameters", Internet draft – work in progress, March 2001, see: www.ist-tequila.org/sls.html

[Hop00] C. Hopps, "Analysis of an Equal -Cost Multi-Path Algorithm," RFC 2992, November 2000

[Li98] T. Li and Y. Rekhter, "A Provider Architecture for Differentiated Services and Traffic Engineering (PASTE)", RFC 2430, Informational, October 1998

[Moy98] J. Moy. "OSPF Version 2", RFC 2178, Standards Track, April 1998

[Nich99] K. Nichols, V. Jacobson and L. Zhang "A Two-bit Differentiated Services Architecture for the Internet", RFC 2638, Informational, July 1999

[Ros01] E. Rosen, A. Viswanathan, and R. Callon, "Multi-Protocol Label Switching Architecture", RFC 3031, Standards Track, January 2001

[Shai99] A. Shaikh et al. "Load-Sensitive Routing of Long-Lived IP Flows", in Proc. ACM SIG-COMM, Cambridge, MA, September, 1999

[Srid01] A. Sridharan et al. "On the Impact of Aggregation on the Performance of Traffic Aware Routing", in Proc. IEEE INFOCOM, Alaska, 2001

[Tha00] D. Thaler and C. Hopps, "Multipath Issues in Unicast and Multicast", RFC 2991, Informational, November 2000

[Trim01] P. Trimintzios et al. "A Management and Control Architecture for Providing IP Differentiated Services in MPLS-based Networks", to appear in IEEE Communications Magazine, May 2000

[Wan99] Y. Wang and L. Zhang, "On the routing equivalence of OSPF and MPLS for IP traffic engineering" Bell Labs Technical Memorandum, May 1999