A CASE STUDY OF MULTISERVICE, MULTIPRIORITY TRAFFIC ENGINEERING DESIGN FOR DATA NETWORKS

Debasis Mitra K. G. Ramakrishnan

Bell Laboratories, Lucent Technologies
Murray Hill, NJ 07974

Abstract

We present techniques, some of which are novel, for traffic engineering in QoS-supported data networks, and also illustrate the application of these techniques in a case study. In the interest of scalability all these techniques use multicommodity flow (MCF) solution techniques as primitives. The techniques address the design of topology and size of explicit routes in MPLS-supported IP networks and VPNs. The techniques are for network-wide optimization subject to constraints on routing imposed by end-to-end QoS and other considerations. The notion of admissible route sets specific to service class and source-destination pair is used to differentiate QoS constraints on real-time services, such as Internet telephony and video, and relatively delay insensitive services, such as premium data. Contrasting optimization techniques are given for services, such as Best Effort, for which no restriction on routes are imposed. Another important traffic engineering requirement addressed is priorities, for which an efficient and accurate design technique is given.

1 Introduction

This paper presents techniques, some of which are novel, for traffic engineering in emerging QoS-supported data networks. The case study serves as a medium for illustrating the techniques. An important property required of the techniques is scalability, so that networks ranging from tens to hundreds of nodes and many service classes may be handled. For this reason all the techniques described here use as primitives the techniques for solving MultiCommodity Flow (MCF), and, more generally, Linear Programming (LP) problems [1, 2]. These primitives are polynomial time algorithms, and the user may select from a large range of options, including off-the-shelf packages for moderate sized problems [3], interior point methods for large problems [4, 5], and algorithms based on combinatorial optimization and ϵ-approximation for even larger problems [6, 7, 8, 9]. Speed of response is important since we expect these traffic engineering tools to be used on-line for batches of demands, and on a semi-on-line basis to perform reconfigurations for larger batch sizes. The approach taken in this paper differs fundamentally from traffic engineering tools for capacity planning which, in the pursuit of global optimization and accuracy, deals with large stochastic models and are necessarily off-line. See [10] and references therein for examples. Stochastic variability is, of course, an important component of traffic engineering for data networks. Packet level variability is taken into account in our flow based approach primarily through the notion of effective bandwidth of flows.

A major incentive for the work here is the emergence of traffic engineering support in IP networks, for instance in networks supporting the MPLS protocol. The task of selecting the topology and size of MPLS "explicit routes" to carry QoS-homogeneous traffic is addressed here. MPLS supports multiple paths from source to destination since this offers the potential of better network-wide load balancing, and this point of view fits well with that of the present work. Also it has been observed that it is mutually advantageous to combine DiffServ and MPLS protocols by carrying ToS-homogeneous traffic on explicit routes [11].

QoS-centric Virtual Private Networks (VPN) introduce additional complexities in traffic engineering. The number of policy and end-to-end constraints in the network proliferate when requirements on routing are introduced by individual VPNs, and the number of VPNs is large, as is expected to be the case. The main point is that the techniques described here for handling QoS constraints are naturally extendible to handle the requirements introduced by VPNs. The size of the problem is larger and the need for scalable techniques is even greater.

This paper develops and illustrates techniques for network-wide optimization subject to constraints on routing imposed by QoS and other considerations. A major motivation is the realization that the converged data networks will support Internet telephony as well as packetized video and other delay sensitive services, in addition to data in a variety of grades of services. To handle such end-to-end constraints we rely on the notion of *admissible route sets*, $\mathcal{R}(s, \sigma)$, which are specific to each QoS service class, s, and (source, destination) pair σ. For instance, real-time services, such as voice and video, may have admissible routes which are severely restricted in their number of hops, while for a relatively delay insensitive service class, such as premium data, the hop restriction may be less stringent. Admissibility of a route will additionally depend on policy, nodal ability to handle certain services, etc., making the generation of route sets a substantial task in itself. We present techniques that handle routing constraints embedded in the admissible route sets. We also present a contrasting technique which is far more efficient in the absence of such constraints, as in the service class in our case study which we call “Best Effort”.

Another important aspect of traffic engineering is priorities. In the case study only a single level of priority is considered; however, the approach naturally extends to multiple priority levels. In the case study priority access to bandwidth is given to the QoS service classes over Best Effort traffic. As a consequence the performance of the QoS service classes is oblivious to the presence of the Best Effort traffic. The design problem with priorities is formulated as a multicriterion optimization problem, where the first objective is the maximization of a weighted combination of QoS-assured carried bandwidth of the QoS service classes, and the second objective is to maximize a similar measure of the Best Effort traffic subject to the constraint that the measure for the QoS traffic is optimal. It is computationally burdensome to obtain an exact solution for this problem. For this reason we develop an efficient and accurate approximation based on a three-layered decomposition of the problem.

An important topic not treated here for lack of space, but considered in [12] is a technique for obtaining traffic engineering designs for stochastic models and environment by adapting MCF-based designs, with only a small amount of incremental effort. In the stochastic model that we consider the basic entities are flows or calls, which arrive as a stochastic point process and persist for randomly distributed time periods, and if accepted in the network, require an (effective) bandwidth which is characteristic of the burstiness and QoS requirements of their service class. In our technique [12] the multicommodity flow rates, which are inputs to the MCF-based design problem, are inflated to compensate for the variabilities in the stochastic model.

The case study is for a small network with 8 nodes and 20 links, which is convenient because results for this network may be readily reproduced by the interested reader, and also prior results are available for comparison. All the results reported here have been obtained from the software package TALISMAN [13], which has been extended to automate the procedures described here and has been exercised for networks substantially larger than the one described here.

2 Model, Problem Formulation and Procedures

We first describe the model of the services, traffic and routes in the network, then describe the problem formulation, and finally give the solution procedures. The results obtained by these procedures for the case study are reported in subsequent sections.

2.1 Model

We consider a connected network with N nodes and L directed links, where link ℓ has bandwidth or capacity C_ℓ. The service classes are classified into QoS and Best Effort. Typically, as in the present case study, there are several QoS service classes, each having characteristic specifications, such as on delay. Here we assume that there is a single Best Effort service class. The QoS group of service classes is denoted by $\mathcal{S}_{\mathrm{QoS}}$.

For each service class s we assume that there is a N by N matrix $\boldsymbol{T}_s$ of bandwidth demands from source node σ_1 to destination node σ_2, i.e., $\boldsymbol{T}_s = \{T_{s,\sigma}\}$. We will find it convenient to refer to (s,σ) as a stream, and to (s,r) as a service-route, where r is a route.

The objectives to be maximized are *network revenues*, which has two components, W_{QoS} and W_{BE} for the QoS and Best Effort service class groups, respectively. Since the former is given priority, comparisons between the two are not particularly meaningful; however, it is still useful to compare the measures from one design to another. We let

$$W_{\mathrm{QoS}} = \sum_{s \in \mathcal{S}_{\mathrm{QoS}}} \sum_{\sigma} \sum_{r \in \mathcal{R}(s,\sigma)} e_{sr} X_{sr} \qquad (2.1)$$

where X_{sr} is the flow or carried bandwidth on service-route (s,r), and e_{sr} is the corresponding earnings per unit carried bandwidth. A special case of particular importance is $e_{sr} = e_{s\sigma}$, for all $r \in \mathcal{R}(s,\sigma)$. The earnings parameter may also be used to tune QoS. Also,

$$W_{\mathrm{BE}} = \sum_{\sigma} e_{\mathrm{BE},\sigma} F_{\sigma} \qquad (2.2)$$

where F_σ is the total Best Effort flow or carried bandwidth for (source, destination) pair σ, and $e_{\mathrm{BE},\sigma}$ is the corresponding earning rate. In agreement with the previously stated assumptions on routing of Best Effort traffic, the earning rate does not depend on the route.

2.2 Formulation of the Traffic Engineering Problem

We give the formulation in three pieces, first for the QoS traffic in isolation, second for Best Effort traffic also in isolation, and finally the combined problem.

2.2.1 Admission Control/Routing for QoS Traffic

The problem is

$$W^*_{\text{QoS}} = \max_{\{X_{sr}\}} W_{\text{QoS}} \tag{2.3}$$

subject to

$$\left.\begin{array}{ll} \sum_{r \in \mathcal{R}(s,\sigma)} X_{sr} \le T_{s,\sigma} & (2.4) \\ X_{sr} \ge 0 \;\; \forall r \in \mathcal{R}(s,\sigma) & (2.5) \end{array}\right\} \forall s \in \mathcal{S}_{\text{QoS}},\ \forall\sigma$$

$$\sum_{s \in \mathcal{S}_{\text{QoS}}} \sum_{\sigma} \sum_{r \in \mathcal{R}(s,\sigma):\ell \in r} X_{sr} \le C_\ell \qquad \forall \ell \tag{2.6}$$

Here X_{sr} is the flow in service-route (s, r) and the slack in (2.4) represents the loss, which may be ascribed to admission control, in stream (s, σ). The ratio of this loss to $T_{s\sigma}$ may be called the loss ratio, $L_{s\sigma}$.

2.2.2 Admission Control/Routing for Best Effort Traffic

The problem is

$$\max_{\{Y_{\sigma\ell}\}} W_{\text{BE}} \tag{2.7}$$

subject to

$$0 \le F_\sigma \le T_{\text{BE},\sigma} \qquad \forall\sigma \tag{2.8}$$

$$\sum_{\ell \in H(n)} Y_{\sigma\ell} - \sum_{\ell \in J(n)} Y_{\sigma\ell} = \left\{\begin{array}{ll} F_\sigma & \text{if } n = \sigma_2 \\ -F_\sigma & \text{if } n = \sigma_1 \\ 0 & \text{otherwise} \end{array}\right\} \begin{array}{l} \forall \text{ nodes } n, \\ \forall\sigma \end{array} \tag{2.9}$$

$$Y_{\sigma\ell} \ge 0 \qquad \forall\sigma, \forall\ell \tag{2.10}$$

$$\sum_{\sigma} Y_{\sigma\ell} \le C_\ell \tag{2.11}$$

In (2.9),

$$H(n) = \text{ set of links directed into node } n, \tag{2.12}$$

and

$$J(n) = \text{ set of links directed out of node } n. \tag{2.13}$$

In the above link-based formulation, $Y_{\sigma\ell}$ represents the component of the total flow F_σ from source σ_1 to destination σ_2 that is carried on link ℓ [14]. Note that no part of the flow F_σ that is admitted into the network is dropped inside the network. As before, the slack $(T_{\text{BE},\sigma} - F_\sigma)$ may be considered the loss due to admission control, and the loss ratio may be obtained by dividing it by $T_{\text{BE},\sigma}$.

The solution to the above problem does not directly shed light on the utilized paths or routes, possibly more than one, from σ_1 to σ_2, and the share of F_σ on these routes. This is addressed separately in a procedure called *flow decomposition*, which involves processing the variables $\{Y_{\sigma\ell}\}$ and $\{F_\sigma\}$ from the solution to the problem in (2.7)–(2.13).

2.2.3 Combined Admission Control/Routing for QoS and Best Effort Traffic

The combined problem is

$$\max\ W_{\text{BE}} \tag{2.14}$$

subject to

$$W_{\text{QoS}} = W^*_{\text{QoS}}\,, \tag{2.15}$$

$$\sum_{s \in \mathcal{S}_{\text{QoS}}} \sum_{\sigma} \sum_{r \in \mathcal{R}(s,\sigma):\ell \in r} X_{sr} + \sum_{\sigma} Y_{\sigma\ell} \le C_\ell \;\forall\ell \tag{2.16}$$

and (2.4), (2.5), (2.8)–(2.10). In (2.15) W^*_{QoS} is obtained from (2.3), i.e., by solving the problem for the QoS traffic in isolation, which has been formulated in Section 2.2.1. Hence, a tacit assumption in the above formulation is that the QoS problem is solved first, followed by the combined problem. Note that the variables $\{X_{sr}\}$, which are common to both problems, may have different values. That is, the only coupling between the two problems is expressed in (2.15).

The combined problem is substantially more complex than either of the problems for the QoS or Best Effort traffic. We give below an approach which decomposes the combined problem and reduces the complexity to the same order as the individual QoS and Best Effort problems, and yet is very effective in approximating the optimal solution. For our case study we compute and report on the exact solution to the combined problem as well as that obtained by the approximation based on the decomposition.

2.3 Multilayered Decomposition of the Combined Problem

In this approach we decompose the combined problem in (2.14)–(2.16) into three layered sub-problems. The first layer simply handles the route-based admission control/routing for the QoS classes, as described in Section 2.2.1, in isolation. The second layer, which requires the solution of a LP of the same order of complexity as the one in the first layer, undertakes resource conservation by minimizing the bandwidth-hops utilized by the QoS traffic, while satisfying the constraint that the earned revenue is optimal, i.e., $W_{\text{QoS}} = W^*_{\text{QoS}}$. An output of the procedure in layer 2 is the residual or unutilized bandwidth C'_ℓ on each link ℓ, which collectively define the residual network $\mathcal{N}'$. In layer 3 the Best Effort admission control/routing problem specified in Section 2.2.2 is solved with C_ℓ replaced by C'_ℓ in (2.11), i.e., the problem is solved on the residual network.

The resource conservation procedure in layer 2 is redundant if there is a unique admission control/routing solution from layer 1 for the QoS traffic. In fact, however, as a result of a phenomenon in typical practical designs, which is well rationalized by a scaling argument from stochastic loss networks but not developed here, the optimal solutions of layer 1 exist in a multidimensional subspace. In fact, in this paper we will provide extensive insights into the structure of the range of possible solutions from layer 1. To this end we employ the tactic of examining the solution of layer 1 which *maximizes* resource usage in bandwidth-hops by the QoS traffic, while also satisfying the constraint $W_{\mathrm{QoS}} = W^*_{\mathrm{QoS}}$. This solution and the solution from layer 2 are, in some sense, the corner points of the range.

A consequence of our multilayered approach is that the residual network $\mathcal{N}'$ on which the Best Effort traffic operates has the property that the sum of its link bandwidths over all links is maximum subject to the constraint $W_{\mathrm{QoS}} = W^*_{\mathrm{QoS}}$. Yet such a maximal residual network may have undesirable characteristics. For instance, it may isolate source nodes from which high earning Best Effort traffic originates or terminates. It is this aspect that the combined problem in Section 2.2.3 addresses and is overlooked in our decomposed solution procedure.

2.4 Route and Link Based Formulations

We draw attention to the differences in the formulations and the variables, $\{X_{sr}\}$ and $\{Y_{\sigma\ell}\}$, in the QoS and Best Effort problem formulations, respectively. The former is route-based, while the latter is link-based [2, 14]. The route-based formulation allows the routes for the different (source, destination) pairs and services to have distinct "policy" constraints. Such routing policy constraints are difficult to incorporate in the link based formulation. On the other hand, when the number of admissible routes is very large the complexity of the route based formulation is correspondingly large. Unless otherwise pruned by policy constraints, this is likely to be the case in richly connected, mesh-like networks where the number of candidate routes may be exponential in the number of nodes. The complexity of the link based formulation depends on the number of links and nodes, but is independent of the number of routes. The above considerations combine to make the route and link based formulations well-matched to the QoS and Best Effort problem layers, respectively.

We next draw attention to the possibility of a further reduction in complexity in the link based formulation of the Best Effort problem layer, at the cost of some loss of generality in the revenue definition, see (2.2). In this formulation only σ_2 would be used to index the flow $Y_{\sigma_2\ell}$ for destination σ_2 on link ℓ. The reduction in the number of variables, and thereby complexity, is clear. The disadvantage is that in the expression (2.2) for revenue, the earnings rate e_{BE,σ_2} may now depend only on the destination. We do not consider such destination and link based formulation any further in this paper.

2.5 Flow Decomposition

We proceed to give a procedure that identifies the utilized routes, i.e., routes with positive Best Effort traffic flows, for each (source, destination) pair σ, from the link flows $\{Y_{\sigma,\ell}\}$ obtained from the layer 3 solution. The following procedure, which is for a particular σ, is repeated for all σ, starting in each case with the residual network $\mathcal{N}'$.

flow-decomposition (σ)

1. reduce graph of $\mathcal{N}'$ by deleting all links ℓ such that $Y_{\sigma\ell} = 0$, and denote it by $\mathcal{G}$
2. procedure trace-route (depth first search):
   trace a route r in $\mathcal{G}$ from source σ_1 to destination σ_2
   compute $X_{\mathrm{BE},r} = \min_{\ell \in r} Y_{\sigma,\ell}$

$$F_\sigma \longleftarrow F_\sigma - X_{\mathrm{BE},r}$$
$$Y_{\sigma,\ell} \longleftarrow Y_{\sigma,\ell} - X_{\mathrm{BE},r} \text{ for all } \ell \text{ in } r$$

3. end if $F_\sigma = 0$
   else goto 1. ■

We mention that an alternative equivalent to the above procedure requires solving a system of linear equations.

3 Network for Case Study

The infrastructure, on which the case study is based, is a fictitious network called Core NSF [13], which has eight nodes ($N = 8$), of which 10 pairs are directly connected, as shown in Figure 1. The network has 20 directed links ($L = 20$), one in each direction for each connected node pair. The typical bandwidth of a directed link is $\mathrm{OC3} = 155$ Mbps, with the exception of the links connecting Argonne 3 and Princeton 4, and also Houston 8 and Atlanta 7, which have bandwidths of $2 \times \mathrm{OC3} = 310$ Mbps. One measure of total resources in the network is 24 OC3-hops.

There are four service classes, of which three (voice, video and premium data, indexed by $s = 1, 2, 3$, respectively) are QoS, and the fourth is Best Effort. Classes 1 (voice) and 2 (video) are real-time, and have identical admissible route sets, namely, only routes with the minimum number of hops from source to destination are admissible. Class 3 is relatively delay-insensitive and its admissible routes have at most 4 hops, which is more than 3, the diameter of the network graph. There are a total of 68 routes in each of the admissible route sets for classes 1 and 2, and 160 routes in the

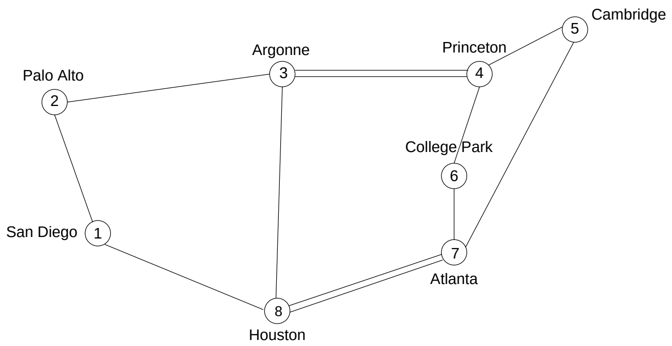

Figure 1: Infrastructure network

set for class 3. As previously stated, there are no restrictions placed on the routes for the Best Effort traffic class.

The point-to-point bandwidth demand for the four services are incorporated in the matrices $\{\boldsymbol{T}_s\}$. For convenience we work with a single base matrix $\boldsymbol{T}$, in which the elements are in units of Mbps, and apply multipliers m_s to obtain $\boldsymbol{T}_s = m_s\boldsymbol{T}$. The multipliers are $m_s = 0.4$, 0.1, 0.25 and 0.25, for $s = 1, 2, 3$ and 4, respectively. The base matrix $\boldsymbol{T}$ is given in Table 1.

									out rates
	–	15.7	18.1	2.7	23.5	13.1	5.3	7.9	86.3
	18.3	–	70.1	8.4	90.4	39.2	15.6	23.5	265.5
	20.8	72.7	–	10.4	96.4	46.8	18.1	26.8	292.0
	2.7	8.4	7.9	–	10.6	5.3	2.7	2.7	40.3
	28.7	96.3	101.6	13.2	–	62.3	23.5	39.2	364.8
	13.1	36.4	41.6	5.3	54.9	–	10.4	15.6	177.3
	5.2	13.1	15.6	2.7	20.8	10.4	–	5.3	73.1
	7.9	20.8	26.0	2.7	31.2	15.6	5.3	–	109.5
in rates	96.7	263.4	280.9	45.4	327.8	192.7	80.9	121.0	1408.8

Table 1: Base matrix $\boldsymbol{T}$, in Mbps

Since classes 1 and 2 have identical route sets and both belong to the QoS group of service classes, they have identical solutions in the flow optimization study, and may be merged. It is only in [12] where we undertake an evaluation of the flow-based design in a stochastic framework, that the two classes have different performances.

Since the sum of all entries in the base matrix $\boldsymbol{T}$ is 1408.8 Mbps, the total demands for the service classes are obtained by applying the appropriate multiplier: 563.5, 140.9, 352.2, 352.2 Mbps.

4 Results from the Case Study

The results of this section are in three parts: first, we give the results for the three QoS service classes; second, the exact and approximate solutions to the combined problem (Sec. 2.2.3) are presented, together with issues and results related to the resource conservation procedure in layer 2 of the decomposition; finally, qualitative features of the solution for the Best Effort service class are discussed.

4.1 Results for the QoS Service Classes

The results below are obtained after completion of layers 1 and 2. The earnings parameter $e_{sr} \equiv 1$, the default setting, so that the earned revenue is identical to the carried bandwidth. The foremost observation on the results is that all bandwidth demands are carried with QoS assurance, i.e., there is no loss. In other words, (2.4) holds with equality. The results are summarized in Table 2.

service class	voice (1)	video (2)	premium data (3)	total
bandwidth demand = carried QoS-assured bandwidth (Mbps)	563.5	140.9	352.2	1056.6

Table 2: Results for QoS service classes

Even though the network resources were not severely strained, as indicated by the no loss condition, the optimal routes that were obtained exhibited some notable departures from their shortest paths. For instance, the Cambridge (5) to Palo Alto (2) traffic is routed as follows: the real-time (classes 1 and 2) streams are routed via Princeton (4) and Argonne (3), as this is the only admissible route for these streams, while the non-real-time (class 3) stream is split, with 78% on the above 3-hop shortest path and the remaining 22% on the longer 4-hop path via Atlanta (7), Houston (8) and San Diego (1). Another interesting case is traffic from College Park (6) to Cambridge. Class 1 and 3 streams are routed via Atlanta, and the class 2 stream is routed via Princeton. Both routes are 2-hop and shortest.

The following features of the residual network $\mathcal{N}'$ are helpful in understanding the routing dispositions for the QoS traffic. The Cambridge to Princeton link has zero capacity, even though the Princeton to Cambridge link has positive capacity. In contrast to the route splitting from Cambridge to Palo Alto, traffic of all QoS classes from Palo Alto to Cambridge are routed on the 3-hop shortest path via Princeton.

4.2 Combined Problem, Resource Conservation

The combined problem in Section 2.2.3 is solved exactly and also by the approximate three-layered technique described in

Section 2.3. We select the earnings parameters $e_{\mathrm{BE},\sigma} \equiv 1.0$, see (2.2), which is the default setting for which the earned revenue W_{BE} is simply the total carried Best Effort bandwidth. The two values of W_{BE} that are obtained are almost identical, with the difference so small as to be plausibly due to numerical errors. The results are in Table 3. Note the heavy usage of the residual network, which together with the uniformity of the earnings parameter constitute conditions in which we expect the approximate solution to be close to the optimum.

total Best Effort bandwidth demand (Mbps):	352.2
W_{BE}, total Best Effort bandwidth carried (Mbps):	279.4
fraction of bandwidth lost (%):	20.5

Table 3: Results for the Combined Problem: exact and approximate solutions

The rest of this section is devoted to the role of layer 2, see Section 2.3, which undertakes conservation of resources utilized by the QoS traffic. As mentioned before, for comparative illustration we obtain the solution that maximizes resource usage, measured in bandwidth-hops, while carrying all QoS traffic demands, just as in the layer 2 solution. The results for these two solutions are in Table 4. Note, by using the result in Table 2, that the average number of hops in routes are about 2.0 and 2.5 in the two solutions, respectively.

Another view of the efficiency of resource conservation is obtained by examining the residual network $\mathcal{N}'$. Let cap($\mathcal{N}'$) denote the total capacity of $\mathcal{N}'$, which is obtained by summing all its link capacities. The results for the two extreme solutions are presented in Table 5. For perspective recall that cap($\mathcal{N}$) $= 24$ OC3.

The effect on connectivity of $\mathcal{N}'$ is noteworthy too. With minimum resource usage by QoS classes only 3 links (Palo Alto $\leftrightarrow$ Argonne, Cambridge $\rightarrow$ Princeton) have zero capacity, while with maximum resource usage 12 links have zero capacity. In the former solution no node is completely isolated, while in the latter solution no Best Effort traffic can leave Palo Alto and Cambridge is isolated in both in and out directions.

minimum resource usage by QoS service classes = 2120 (Mbps-hops)
(layer 2 solution)
maximum resource usage by QoS service classes = 2668 (Mbps-hops)

Table 4: Resource usage by QoS traffic

cap($\mathcal{N}'$) if resource usage by QoS classes is minimized = 10.3 OC3
cap($\mathcal{N}'$) if resource usage by QoS classes is maximized = 2.85 OC3

Table 5: Residual network capacity

We should point out that there is an important hidden cost to optimal resource usage by QoS classes. It has the effect of packing routes tightly, which is the opposite, in a certain sense, of load balancing. On the other hand, maximizing resource usage yields well balanced loads on routes. The effect of this dichotomy is not visible in the present flow-based framework, but shows up in the evaluation of designs in the stochastic framework, which is studied in the next section. Here we mention only that there is an important role for less aggressive resource conservation than is implemented in layer 2, for instance by introducing constraints that require link utilizations not to exceed certain levels, which may either be prespecified or obtained adaptively in the course of the design process.

4.3 Results for the Best Effort Class

We comment on the qualitative nature of the solutions obtained for Best Effort traffic routes from the flow decomposition procedure in Section 2.5. The procedure yields routes with positive flows having as many as 5 hops. Not too surprisingly, there are some extreme features in the optimal solution for the Best Effort class. For instance, all Best Effort traffic from Cambridge to Palo Alto and from Cambridge to College Park is lost, while the fraction of bandwidth from Palo Alto to Cambridge that is lost is 78%. On the other hand, Best Effort traffic due to all node pairs, with the exclusion of the aforementioned and all Best Effort traffic originating at Cambridge, is carried without any loss. Such extreme features may be considered unfair and therefore unacceptable. In this case the flexibility afforded by the earnings parameters $e_{\mathrm{BE},\sigma}$ may be exercised to obtain less extreme solutions. For instance, the default setting of the earnings parameters is obviously responsible for streams in which the source and destination are distant to suffer higher losses. The earnings parameters may be tuned to compensate for such phenomenon.

5 Conclusions

We have presented and illustrated scalable techniques, which use multicommodity flow solution methods as primitives, for handling QoS requirements in the traffic engineering of data networks. Novel techniques have been presented for (a) adapting efficient MCF-based design methods for stochastic models, (b) incorporating priorities in the design by an efficient and accurate approximation based on a three-layered decomposition, and (c) matching the requirements of QoS and Best Effort traffic to the solution techniques, route and edge-based respectively, for their routing/admission control problems. Some topics for further investigation are as follows. There is great interest in evaluating various new and powerful combinatorial optimization and ϵ-approximation techniques for extremely large data networks, especially when the time requirements are severe. Also, the results presented in Section 4 suggest other formulations of the combined (priority) problem in which the incentive for packing the higher priority QoS traffic is not as great, thus allowing greater load balancing. In fact, such modifications are more conveniently made in layer 2 of our three layered decomposition. Similarly, it will be beneficial to investigate techniques for achieving greater fairness among (source, destination) pairs by tuning the earnings parameters, i.e., by increasing earnings for more widely separated terminal nodes. Yet another topic for further investigation is the comparative performance of sequential optimization where demands are routed one at a time, in contrast to the joint optimization of batches of demands as discussed here.

References

[1] D. G. Luenberger, *Linear and Nonlinear Programming*. Reading, MA: Addison-Wesley, 2nd ed., 1984.

[2] D. Bertsekas and R. Gallager, *Data Networks*. Englewood Cliffs, NJ: Prentice Hall, 2nd ed., 1992.

[3] B. A. Murtagh and M. A. Saunders, "MINOS 5.1, User's Guide," Tech. Report SOL83-20R, Systems Optimization Laboratory, Dept. of Operations Research, Stanford University, 1983, revised 1987.

[4] I. Adler, N. K. Karmarkar, M. G. C. Resende, and G. Veiga, "Data structures and programming techniques for the implementation of Karmarkar's algorithm," *ORSA J. Computing*, vol. 1, Spring 1989.

[5] N. K. Karmarkar and K. G. Ramakrishnan, "Computational results of an interior point algorithm for large scale linear programming," *Mathematical Programming*, vol. 52, pp. 555–586, 1991.

[6] C. Stein, *Approximation Algorithms for Multicommodity Flow and Scheduling Problems*. PhD thesis, MIT, 1992.

[7] N. Garg and J. Konemann, "Faster and simpler algorithms for multicommodity flow and other fractional packing problems," *Proc. IEEE Symp. on Foundations of Computer Science*, 1998.

[8] R. Hassin, "Approximation schemes for the restricted shortest path problem," *Math. of Oper. Res.*, vol. 17, pp. 36–42, February 1992.

[9] A. Goel, D. Kataria, D. Logothetis, and K. G. Ramakrishnan, "An efficient algorithm for constraint-based routing in data networks," Preprint, 1999.

[10] D. Mitra, J. A. Morrison, and K. G. Ramakrishnan, "Virtual private networks: Joint resource allocation and routing design," *Proc. IEEE INFOCOM*, 1999.

[11] L. Wu, P. Cheval, and P. Vaananen, "MPLS extensions for Differential Services," IETF draft-wu-mpls-diff-ext-00.txt, work in progress, November 1998.

[12] D. Mitra and K. G. Ramakrishnan, "A case study of multiservice, multipriority traffic engineering design for data networks," Bell Labs report, September 1999.

[13] D. Mitra, J. A. Morrison, and K. G. Ramakrishnan, "ATM network design and optimization: A multirate loss network design," *IEEE/ACM Trans. Networking*, vol. 4, no. 4, pp. 531–543, 1996.

[14] S. Ahuja and J. Orlin, *Network Flows*. Reading, MA: Addison-Wesley, 1998.