This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the ICC 2007 proceedings.

TDM Emulation in Packet-Switched Networks

George N. Rouskas, Nikhil Baradwaj
Department of Computer Science, North Carolina State University, Raleigh, NC 27695-8206

Abstract— **Many network operators offer some type of tiered service, in which users may select only from a small set of service levels (tiers). In this work, we study the problem of designing a tiered-service network that allocates bandwidth in multiples of a basic bandwidth unit. Such a packet-switched network can enjoy many of the benefits, in terms of control and management, of a TDM network, but without the associated data plane rigidities.**

I. Introduction

Many networks offer some type of *tiered service*, in which users may select only from a small set of service *tiers* (levels). The main motivation for such a service is to simplify a wide range of core functions (including network management and equipment configuration, traffic engineering, service level agreements, billing, and customer support), enabling the providers to scale their operations to millions of customers. Currently, service tiers are either based on the bandwidth hierarchy of the underlying network infrastructure (e.g., DS-1, DS-3, OC-3, etc.), or are determined in some *ad-hoc* manner (e.g., the various ADSL tiers available through different providers).

The problem of bandwidth quantization, i.e., optimally determining the set of service levels, was studied in [5] in the context of ATM networks, and a heuristic based on simulation annealing was presented. We have recently developed a theoretical framework for reasoning about and tackling algorithmically the problem of service tier selection by formulating the general problem of traffic quantization as a directional p-median problem [1], [3], [4], and we have developed efficient algorithms for a number of important variants.

In this paper, we study a special version of the problem in which all service tiers are multiples of a basic bandwidth unit. A packet-switched network operating with such a set of service levels would resemble a TDM network that allocates bandwidth in multiples of a slot. Consequently, many robust network management functions developed for telecommunication networks, including admission control, routing, traffic engineering and grooming, etc., could be easily adapted for the tiered-service packet-switched network. We emphasize that this "TDM emulation" only concerns the control and management functions, *not* the data plane operation of the network. For example, while bandwidth is allocated in multiples of the basic unit, flows are not limited to using a particular slot. Similarly, unlike TDM networks where an unused slot is wasted, excess bandwidth can be allocated to active flows by the scheduling algorithm. Furthermore, the bandwidth unit is not fixed or determined by hardware, as in a TDM network, but, it is configurable and can be optimized for the characteristics of the carried traffic. In addition, the routers provide for free the functionality of a time-slot interchange. Overall, using TDM emulation, packet-switched networks (e.g., those employing 1 or 10 GE links) can enjoy many of the benefits, in terms of control and management, of a TDM network, but without the data plane rigidities of such a network. Furthermore, this problem has important applications to next-generation SONET networks in which it is possible to use virtual concatenation to allocate bandwidth flexibly in any multiple of 64 Kbps [2].

This work was supported by the NSF under grant CNS-0434975.

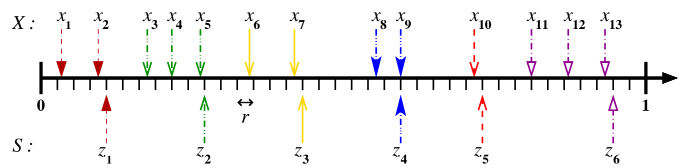

Fig. 1. User requests x_i mapped to service tiers z_j which are multiples of r

In Section II we formulate the problem of service tier selection, and present an exhaustive optimal algorithm. In Section III we develop a suite of efficient algorithms to select both the service levels and the bandwidth unit that are jointly optimal. We present numerical results in Section IV, and we conclude the paper in Section V.

II. Service Tier Optimization

We consider a packet-switched network with n users. Let x_i be the amount of bandwidth requested by user i. Let r denote the unit bandwidth, i.e., the smallest quantity in which the network allocates bandwidth. Without loss of generality, we assume that quantities r and $x_i, i = 1, \cdots, n$, are expressed as fractions of the maximum link capacity C in the network, i.e., $0 < r, x_i \le 1$. The network offers $p \ge 1$ levels (tiers) of service; typically, $p \ll n$. The j-th level of service corresponds to bandwidth $z_j = rk_j$, where k_j is an integer and $z_1 < z_2 < \ldots < z_p \le 1$. In such a tiered-service network, each user i is mapped to service level z_j such that $z_{j-1} < x_i \le z_j$. The additional bandwidth $z_j - x_i$ represents the performance penalty associated with the tiered service.

Let $X = \{x_1, \ldots, x_n\}$ be a set of n points on the real line, such that $x_1 \le x_2 \le \cdots \le x_n$, and define the *density* of X to be $\rho_X = \sum_{i=1}^{n} x_i$. A set of service tiers $S = \{z_1, \ldots, z_p\}$, $z_1 < z_2 < \cdots < z_p$, $1 \le p \le n$, is a *feasible solution* for X if and only if $x_n \le z_p$. Associated with a feasible solution is an *implied mapping* $X \to S$, where $x_i \to z_j$ if and only if $z_{j-1} < x_i \le z_j$. Figure 1 shows a sample mapping from a set of X of 13 demand points onto a set S of 6 service tiers which are multiples of some unit r.

Let n_j denote the number of requests mapped to tier z_j. We define the service tier optimization (STO) problem as follows.

Problem 2.1 (STO): Given a set X of n bandwidth requests, $x_1 \leq x_2 \leq \cdots \leq x_n$, and a constant B, find a real r and a feasible set S of p service tiers, $z_1 < z_2 < \cdots < z_p$, $1 \leq p \leq n$, so as to minimize the objective function:

$$Obj(S,r) = \left[\sum_{j=1}^{p}(n_j z_j) - \rho_X\right] + \frac{B}{r} \qquad (1)$$

under the constraints: $z_j = rk_j,\ k_j$: integer, $j = 1, \ldots, p$.

The objective function above consists of two terms which represent a tradeoff with respect to the selection of the bandwidth unit r. The density ρ_X is the amount of bandwidth requested, while $\sum_{j=1}^{p}(n_j z_j)$ is the bandwidth assigned to the users under the tiered service. Hence, the term in the brackets in (1) is the amount of excess bandwidth needed by the tiered-service network to accommodate the set X after mapping it to the service tier set S. It is not difficult to see that the term in brackets is minimized when $r = \frac{1}{C}$, i.e., when bandwidth is allocated at the finest possible granularity[1]: since service tiers must be multiples of r, if bandwidth is allocated at a granularity coarser than $r = \frac{1}{C}$, the solution space for the service tier set $S = \{z_j\}$ will be a proper subset of the solution space when $r = \frac{1}{C}$, yielding a suboptimal solution for the excess bandwidth penalty expressed by the term in brackets.

On the other hand, the term $\frac{B}{r}$ in the objective function (1), where B is some constant related to the operation of the system, prevents r from being very small. Specifically, the term $\frac{B}{r}$ is of practical importance as it captures the overhead associated with making the unit r of bandwidth allocation small. To illustrate, let us make the simplifying assumption that all users request and receive the basic rate of rC bits/sec. After serving a user, the system incurs some overhead due to the bookkeeping operations, memory lookups, etc., required before it can switch to serving another user. Let α denote the amount of time required to switch between users, expressed as the number of bits that could be transmitted during this time at the given service rate. Therefore, the quantity $\frac{\alpha}{rC}$ represents the amount of overhead operations relative to the unit rate. This relative overhead, which increases as the unit of bandwidth decreases, is similar in principle to the "cell tax" incurred in carrying IP traffic over ATM networks due to the relatively large fraction of header (i.e., overhead) bits to data bits. In the objective function (1) we use the term $\frac{B}{r}$ where $B = \frac{c\alpha}{C}$ and c is a constant which ensures that the two terms in the right hand side of (1) are expressed in the same units.

A. Optimal Solution to STO for Fixed r

As defined, the objective of STO is to find jointly optimal values for the basic bandwidth unit r (a real number) and the p service tiers. However, let us consider for a moment the special case where the value of r is fixed and not subject to optimization; as we shall see shortly, the algorithm for this problem is useful in tackling the general one. In this case, the term $\frac{B}{r}$ in (1) is constant and does not affect the minimization.

[1] Recall that r is expressed as a fraction of the link capacity C; therefore, letting $r = \frac{1}{C}$ implies that bandwidth is allocated at a granularity of 1 bit/sec.

Consider an instance of STO in which the value of r is fixed at $r = r'$; that is, the p service tiers can only take the values $kr', k = 1, \ldots, K$. Let $U = \{u_1, \ldots, u_K\}$ be the set of candidate values for the p service tiers, $u_k = kr'$; in Figure 1, these candidate values are represented by the ticks below the horizontal line. Integer K corresponds to the largest possible multiple of r', i.e., $K = \lceil \frac{x_n}{r'} \rceil$, where x_n is the largest demand.

This version of STO is a special case of the directional p-median problem we defined and studied in [4]. Consequently, this problem can be solved in time $O(pK)$ using a dynamic programming algorithm; the reader is referred to [4] for details on the dynamic programming formulation.

B. The Behavior Of The STO Objective Function

To obtain insight into how the unit rate r affects the optimization, in Figure 2 we plot the objective function against the value of r for an instance of STO with $n = 1000$ user requests, $p = 10$ service tiers, and $B = 0.01$, with the set of n requests generated from a uniform distribution in (0,1). We varied the value of the basic unit r in increments of $\delta_r = 10^{-5}$. For each (fixed) value of r we obtained the optimal service tiers using the dynamic programming algorithm we developed in [3], which does not take into account the term $\frac{B}{r}$ of the objective function. Once we obtained the service tiers, we evaluated the objective function (1), including the term $\frac{B}{r}$.

The behavior exhibited in Figure 2 is representative of the STO instances we have studied. At low values of r, the term $\frac{B}{r}$ representing the overhead cost dominates, resulting in large overall values. As r increases, there is an initial period of rapid decrease in the objective function as the term representing the excess bandwidth penalty starts to become important. The curve then settles into a seesaw pattern. The high and low points along this pattern depend on the values of the multiples of r relative to the demand points: when multiples of r are aligned close to user requests, there is little bandwidth penalty for mapping these requests to service tiers that are multiples or r, hence the objective function has a lower value; the opposite is true when there is a mismatch between multiples or r and user requests. Also, as the value of r increases further, the curve trends upwards. This behavior is due to two factors that come into play when r becomes large: the excess bandwidth term (i.e., the one in brackets) in (1) starts to dominate, and at the same time this term increases in value as large values of r are too coarse to minimize the excess bandwidth.

C. An Exhaustive Search Algorithm for STO

It is clear from Figure 2 that the objective function is non-convex and includes several troughs at irregular intervals. This non-convex nature makes standard optimization techniques (e.g., steepest descent methods) impractical, as it is very easy to get trapped in a local minima. We now describe an exhaustive search approach for identifying the value of r and the service tiers that minimize the objective function, and in the next section we develop a suite of heuristics that trade solution quality for running time.

For a STO instance with p service tiers and x_n the largest user request, the largest value that r may take under the

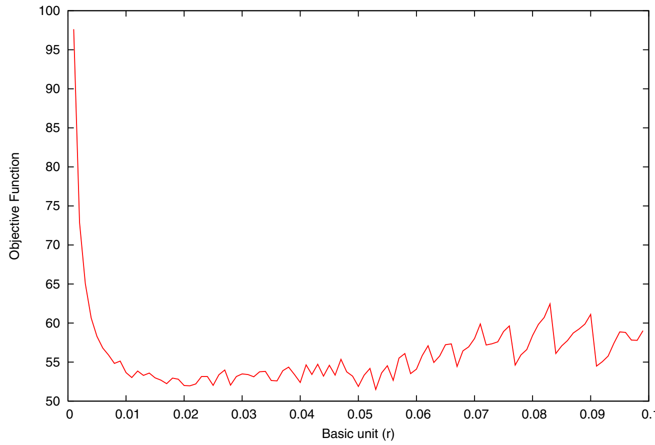

Fig. 2. Objective function value against r, $n = 1000$, $p = 10$, $B = 0.01$, demand points generated from a uniform distribution in $(0, 1)$

constraint that all p service tiers be an integer multiple of r is $r_{max} = \frac{x_n}{p}$. Hence, the optimal value of r lies in the interval $(0, r_{max}]$. Let δ_r be a small increment value, and consider the set $R = \{r_m = m\delta_r \leq r_{max}, m = 1, 2, \ldots\}$. For each (fixed) $r_m \in R$, we use the approach in Section II-A to obtain the optimal service tiers, and evaluate the objective function (1). The optimal solution to the STO problem is obtained as the value of r_m and the corresponding service tiers which produce the smallest value for the objective function.

In order to determine the running time complexity of this exhaustive search algorithm, let $L = \lfloor \frac{r_{max}}{\delta_r} \rfloor$ be the size of set R (i.e., the number of candidate values of r to be considered), and $K_m = \frac{x_n}{r_m}$ be the number of candidate service tiers when the value of $r = r_m$. The dynamic programming algorithm will be run L times, and during the m-th iteration, i.e., when $r = r_m$, the algorithm will take $O(pK_m)$ time. Since

$$\sum_{m=1}^{L} pK_m = \frac{px_n}{\delta_r} \sum_{m=1}^{L} \frac{1}{m} = \frac{px_n}{\delta_r}(\ln L + \gamma) \qquad (2)$$

where $\gamma = 0.577\ldots$, is Euler's constant, the complexity of the algorithm is $O(\frac{px_n}{\delta_r} \ln(\frac{x_n}{p\delta_r}))$, and depends critically on the value of the increment δ_r which determines the granularity of the search. With finer granularity (i.e., smaller δ_r), the accuracy of the algorithm increases, but its complexity also increases dramatically; the opposite is true when δ_r becomes larger and the granularity coarser. Also, the time complexity is independent of the number n of users, and depends only on the largest demand x_n which is bounded above by the bandwidth available on the highest capacity link in the network. Consider a network with 10 Gbps links. A reasonable value for the increment is $\delta_r = 64$ Kbps. Assuming that the largest demand can be equal to the capacity of a link, we have that $\frac{x_n}{\delta_r} \approx 10^6$, which demonstrates that the exhaustive search is taxing in terms of both computational and memory requirements.

III. Optimization Heuristics

We now present a set of heuristics for the STO problem. Each heuristic trades solution quality for speed by using its own approach to reduce the size of the space of candidate values for r and/or the service tiers that it considers.

A. Demand Driven Heuristic (DDH)

Recall that, for each candidate value r_m of r, the exhaustive search algorithm considers all the $K_m = \frac{x_n}{r_m}$ multiples of r_m as the set of potential service tiers, where K_m can be much larger than the number n of user requests. The intuition behind this heuristic is that the optimal service tiers are more likely to be located just above a user request, since otherwise there would be a larger penalty in terms of excess bandwidth. Therefore, the heuristic only considers the n multiples of r_m that are located immediately to the right of (or coincide with) the n user demands. In other words, the set U of candidate values for the p service tiers is $U = \{u_i = r_m \times \lceil \frac{x_i}{r_m} \rceil, i = 1, \ldots, n\}$. Since there have to be n different candidate service tiers, the range of values for r is in the interval $(0, \frac{x_n}{n} = r_{max}]$. Using n instead of K_m and the new value for r_{max} in expression (2), we find that the running time complexity of the DDH heuristic is $O(\frac{px_n}{\delta_r})$, which represents an improvement over the exhaustive algorithm, especially for small values of δ_r which allow for a finer granularity search.

B. Service Tier Driven Heuristics

Both the DDH and the exhaustive search algorithms apply the dynamic programming algorithm in [3] for each candidate value for parameter r. The two heuristics we present in this section are based on the assumption that the optimal service tiers for STO are likely to be close to the optimal service tiers for the corresponding problem with the same user demand set X but in which $r = r_0 = \frac{1}{C}$. Therefore, each heuristic initially runs the dynamic programming algorithm in [3] and computes the optimal set $S^{r_0} = \{z_1^{r_0}, \ldots, z_p^{r_0}\}$ of service tiers for this problem. This step takes time $O(pn)$, and this dynamic programming algorithm is not used again by the heuristics.

The first algorithm, which we call the *unidirectional service tier driven heuristic (USDH)*, sets the i-th service tier for a given candidate value r_m of r to the smallest multiple of r_m that is greater than or equal to service tier $z_i^{r_0}$. In other words, the set S_m of service tiers for candidate r_m is defined as $S_m = \{\lceil \frac{z_i^{r_0}}{r_m} \rceil r_m, i = 1, \ldots, p\}$. The heuristic returns the value r_m and corresponding set S_m which result in the minimum value for the objective function (1).

The second algorithm is called the *bidirectional service tier driven heuristic (BSDH)*, and computes a set of $2p$ possible values for the service tiers for each candidate value r_m. The first set of p values is identical to the set S_m used by the USDH algorithm above. In addition, this heuristic considers the set S'_m consisting of the p largest multiples of r_m that are less than the corresponding service tiers $z_i^{r_0}$, i.e., $S_m = \{\lfloor \frac{z_i^{r_0}}{r_m} \rfloor r_m, i = 1, \ldots, p\}$. The $2p$ elements of these two sets collectively become the candidates for being one of the p service tiers when the value of $r = r_m$. We use the dynamic programming algorithm in [3] to select the optimal set of p service tiers from the set $S_m \bigcup S'_m$. As with USDH, the heuristic returns the value r_m and corresponding p service tiers that minimize (1).

We expect the BSDH heuristic to perform better than USDH since it considers a larger number of candidate service tiers;

TABLE I
FORMULAE FOR THE PDF AND CDF OF THE INPUT DISTRIBUTIONS

Distribution	pdf	cdf	Domain
Uniform	1	x	$0 \le x \le 1$
Increasing	$2x$	x^2	$0 \le x \le 1$
Decreasing	$-2x+2$	$-x^2+2x$	$0 \le x \le 1$
Triangle	$4x$	$2x^2$	$0 \le x < 0.5$
	$-4x+4$	$-2x^2+4x-1$	$0.5 \le x \le 1$
	4/9	$4x/9$	$0 \le x < 0.25$
Unimodal	6	$6x-25/18$	$0.25 \le x < 0.35$
	4/9	$4x/9+5/9$	$0.35 \le x \le 1$
	1/4	$x/4$	$0 \le x < 0.25$
	4	$4x-15/16$	$0.25 \le x < 0.35$
Bimodal	1/4	$x/4+3/8$	$0.35 \le x < 0.65$
	1	$4x-33/16$	$0.65 \le x < 0.75$
	1/4	$x/4+3/4$	$0.75 \le x \le 1$

this improved performance, however, is at the expense of having to run the dynamic programming algorithm on a set of $2p$ values, which takes time $O(p^2)$.

C. The Power of Two Heuristic (PTH)

This heuristic simply selects the set of p service tiers as the set of the p consecutive powers of two such that the largest element in the set is the smallest power of two that is larger than or equal to the largest user request x_n; in other words, $S = \{2^{q+1}, 2^{q+2}, \ldots, 2^{q+p} \mid 2^{q+p-1} < x_n \le 2^{q+p}\}$. This solution is consistent with STO in that it consists of service tiers all of which are a multiple of a basic unit, in this case 2^{q+1}. However, as we shall see shortly, the excess bandwidth penalty for this solution can be quite high compared to the other algorithms. We consider this solution here as a baseline case as it is similar in spirit to approaches that assign packet flows in classes (e.g., as in [6]) whose boundaries are defined by powers of two.

IV. NUMERICAL RESULTS

We now present simulation results to investigate the relative performance of the various algorithms we presented and to determine their effect on the operation of a network. The demand sets X of the problem instances we consider throughout this section were generated from one of six distributions whose pdf and cdf are listed in Table IV. Recall that that bandwidth demands are normalized with respect to the link capacity, hence, the domain of the pdf and cdf of all distributions in Table IV is [0,1]. Also, based on our discussion in Section II-B and the fact that the largest demand point $x_n \le 1$, we used an increment value $\delta_r = 10^{-5}$ whenever applicable.

Algorithm comparison. Let us first investigate the relative performance of the various algorithms for STO with respect to the objective function (1). Figure 3 plots the value of the objective function returned by four STO algorithms, DDH, BSDH, USDH, and PTH, against the value of r for a representative problem instance. We also show a lower bound (denoted by "STO-LB") obtained by solving the STO problem after excluding the overhead term $\frac{B}{r}$ in the objective function (1). Consequently, the STO-LB solution is such that $r = \frac{1}{C}$, i.e., the service tiers may take any values (since all bandwidth values are multiples of the smallest possible unit $\frac{1}{C}$), and thus serves as a lower bound for the STO algorithms. Since the STO-LB and PTH solutions do not take parameter r into account, they are shown as horizontal lines in the figure. We note that PTH performs much worse than all other algorithms, confirming our earlier observation that exponential grouping of traffic flows is not an efficient approach; this result is representative of the behavior of PTH, hence, we do not consider this heuristic in the remainder of this section. We also observe that the three algorithms DDH, BSDH, and USDH perform close to the lower bound.

For the results shown in Figures 4 and 5, we have considered thirty problem instances with $n = 100$, $p = 5$, and $B = 0.05$, generated from the increasing and triangle distributions, respectively. The figures plot the best (i.e., lowest over all r) objective function value returned by the DDH, BSDH, USDH, and STO-LB algorithms, for each problem instance; again, the STO-LB solution provides a lower bound for the other three algorithms. The graphs show that, except for a few instances, all three STO algorithms are close to the lower bound. Of the three algorithms, DDH produces the lowest objective function values, followed closely by BSDH. The objective function values returned by USDH are generally higher than those of the DDH and BSDH heuristics, but USDH has a much faster running time. Hence, these results indicate that there is a tradeoff between quality of solution and running time complexity of the algorithms.

Bandwidth penalty due to tiered service. Let us now turn our attention to determining the penalty in terms of excess resources needed due to tiered service. Given a demand set X, a continuous-rate link will use an amount of bandwidth equal to $\rho_X = \sum_i x_i$ to satisfy all the demands in X. A link of a tiered-service network, on the other hand, will in general use more bandwidth, as each demand x_i will be mapped to the next offered level of service. For a network with service levels obtained through the STO algorithm, the amount of bandwidth used is given by the objective function (1) after subtracting the term $\frac{B}{r}$. We use the *normalized bandwidth requirement* metric, defined as the ratio of the amount of bandwidth used by a tiered-service network to the amount of bandwidth ρ_X used by a continuous network, to characterize the bandwidth penalty incurred by a tiered-service network. Figures 6 and 7 plot this metric against the number p of service levels offered by the network. Each point in these curves is the average over 30 different problem instances generated by a uniform distribution; we have obtained similar results for all distributions shown in Table IV.

Figure 6 presents results for two tiered-service scenarios: one in which the service levels are obtained from the STO-LB solution, and one in which they are obtained from the DDH algorithm; DDH is selected as a representative algorithm for the STO problem in which the service levels are all multiples of a basic bandwidth unit r. As we can see, the curve for DDH is above the one for STO-LB. This result is expected, since the STO-LB solution is only concerned with minimizing the excess bandwidth due to tiered service, while the DDH algorithm also has to take into account the constraint that

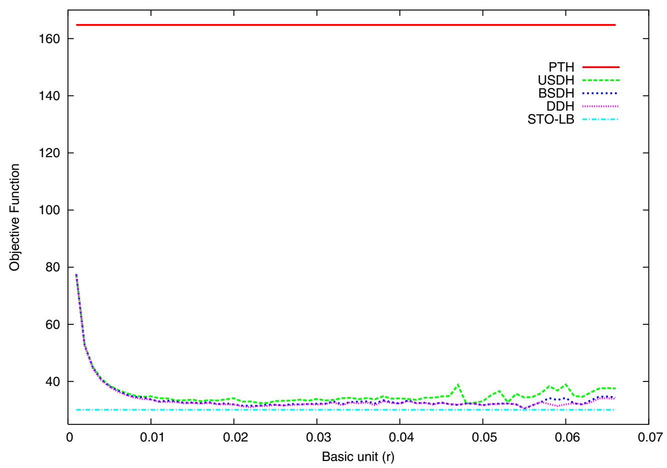

Fig. 3. Objective function value against r, $n = 1000$, $p = 15$, $B = 0.05$, triangle distribution

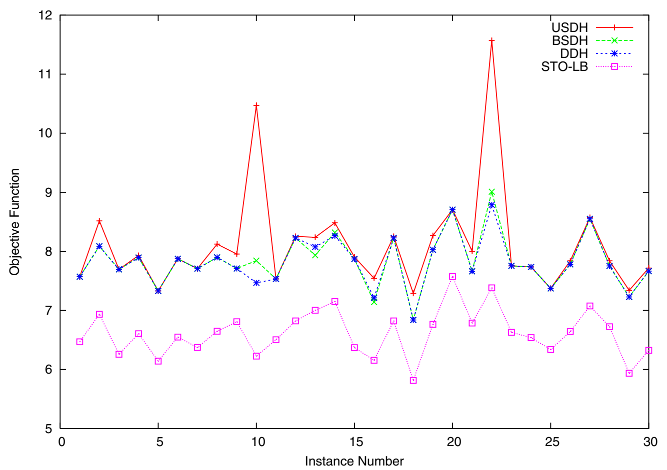

Fig. 4. Algorithm comparison, $n = 100$, $p = 5$, $B = 0.05$, increasing distribution

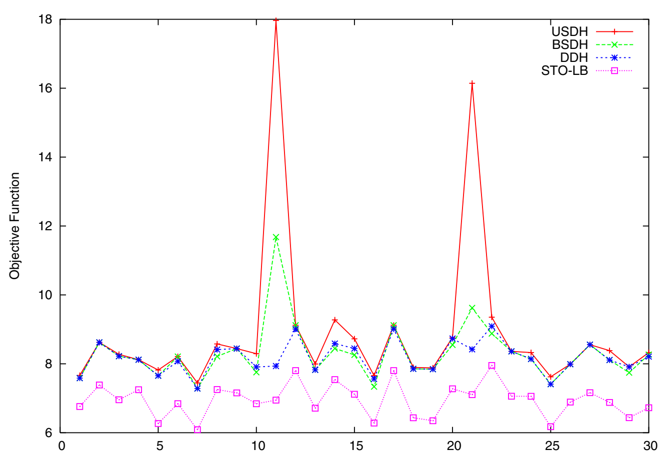

Fig. 5. Algorithm comparison, $n = 100$, $p = 5$, $B = 0.05$, triangle distribution

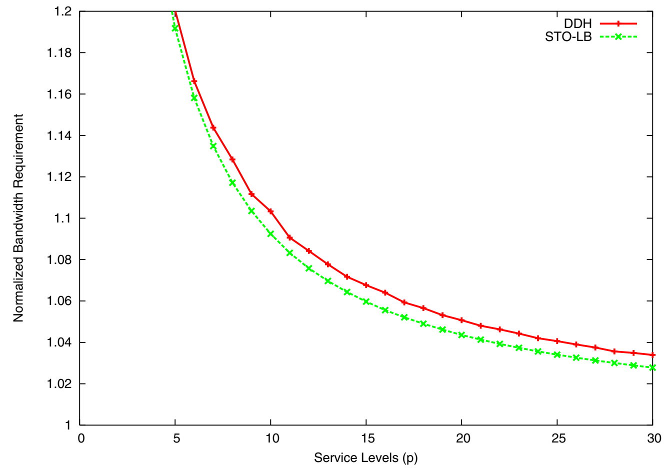

Fig. 6. Normalized bandwidth requirement against p, uniform distribution

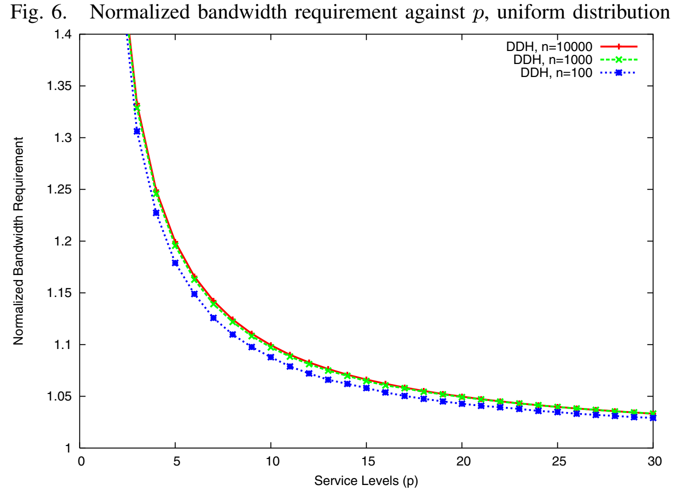

Fig. 7. Normalized bandwidth requirement against p, uniform distribution

all service levels be multiples of a basic unit. However, the additional penalty due to this constraint is relatively small; we have observed similar behavior for all distributions. Also, the normalized bandwidth requirement decreases rapidly with the number of service levels; this result can be explained by noting that as the number of levels becomes very large, the tiered-service network reduces to a continuous-rate network.

Figure 7 shows the effect of the number n of users on the normalized bandwidth requirement for the DDH algorithm; the effect on the STO-LB solution is similar. As the number n of users increases, the normalized bandwidth requirement does increase slightly, but the effect diminishes quickly so that the curve for $n = 10,000$ almost coincides with the curve for $n = 1,000$. The conclusions we can draw from these figures, and similar ones which can be found in [1], is that (1) with $p = 10 - 15$ levels, the bandwidth required by a tiered-service network is only about 5-10% higher than that of a continuous-rate network; (2) the additional constraint that all service levels be a multiple of a basic unit only slightly adds to the bandwidth penalty; and (3) increasing the number n of users imposes only an incremental penalty on bandwidth.

Impact on network performance. Finally, let us examine the practical impact of tiered service on overall network performance. To this end, we consider a scenario in which connections arrive and depart dynamically. A connection between source-destination pair (s, d) requires a certain amount of bandwidth; if a path between s and d with sufficient resources can be found, the connection is established, otherwise, it is rejected (blocked). The performance measure of interest in this context is the connection blocking probability. We use simulation to compare the blocking probability of a continuous network to that of a tiered-service network. In a continuous network, a connection requiring bandwidth x_i is accepted if a path with at least that much bandwidth can be found. In a tiered-service network, the bandwidth demand x_i is first mapped to the next highest service level offered, say, z_j, and the connection is accepted if a path with bandwidth at least equal to z_j is found. The service levels for the tiered-service network are computed in advance for the given demand distribution, as discussed earlier.

In our simulation model, connections arrive as a Poisson process with rate λ, and their mean holding time is an exponentially distributed random variable with rate $\mu = 1$. Each simulation run lasts until 100,000 connection requests have been served. Each point in the blocking probability curves shown here is the average of thirty simulation runs; we also plot 95% confidence intervals which we estimated using the method of batch means. All the results are for the NSFNet network topology, which can be found in [1]. The capacity of all links is set to two units of bandwidth; since the demand distributions in Table IV are defined in the interval $[0, 1]$, this assumption implies that the bandwidth requested by any connection is at most one half the link capacity.

Figure 8 plots the blocking probability against the connection arrival rate for a continuous network and two tiered-service networks, one using the STO-LB solution to obtain the service levels and one using DDH, a representative algorithm for the STO problem. As expected, the blocking probability of the continuous-rate network is lowest, that of the tiered-service network allocating bandwidth in multiples of a basic unit is highest (DDH algorithm), and that of a network (STO-LB solution) which minimizes the excess bandwidth is in between the other two. The higher blocking probability of a tiered-service network is the result of the additional resources that such a network uses for each traffic demand. However, the increase in blocking probability is rather small and it may be more than compensated by the advantages of tiered service.

Figure 9 shows the behavior of the blocking probability for the DDH algorithm as we vary the number of service levels p. The curves confirm the intuition that as p increases, the blocking probability of the tiered-service network decreases and tends towards that of a continuous-rate network. This figure suggests that the network designer/engineer may select the number p of the service levels to be offered so as to combine the advantages of tiered service with the performance of a continuous-rate one.

V. Concluding Remarks

Tiered service has many potential applications in networking, especially in contexts where catering to very large sets

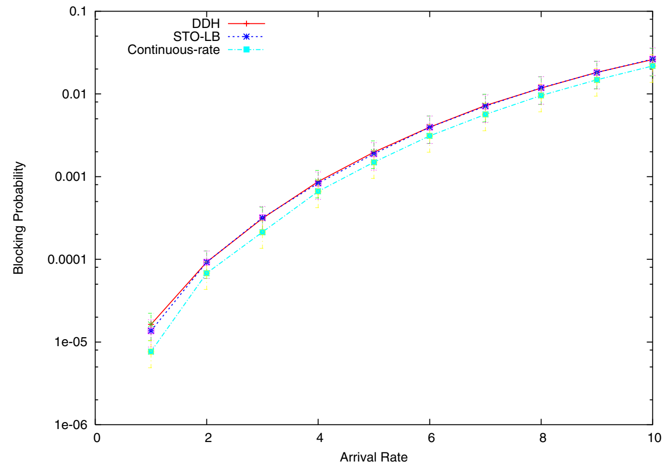

Fig. 8. Blocking probability against the arrival rate, $n = 100,000$, $p = 30$, $B = 0.05$, uniform distribution

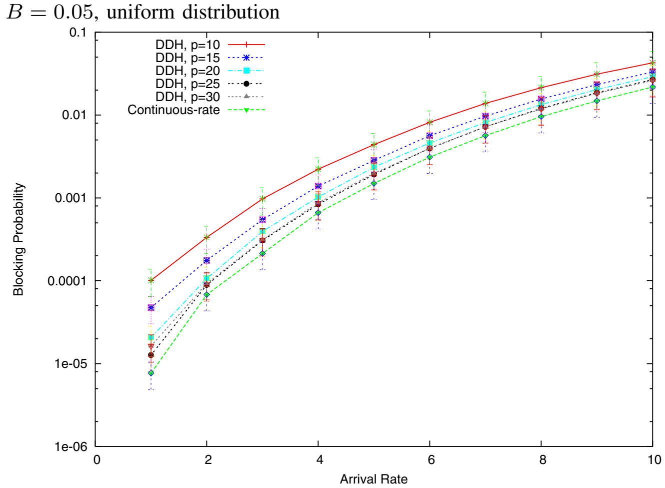

Fig. 9. Blocking probability against the arrival rate, $n = 100,000$, $B = 0.05$, uniform distribution

of heterogeneous users poses significant scalability problems. We have developed efficient algorithms to determine optimal service levels that are multiple of a basic bandwidth unit, allowing packet switched networks to emulate the operation of TDM networks. Our ongoing research aims to extend this work by including pricing considerations in service level selection.

References

[1] Nikhil Baradwaj. Traffic quantization and its application to QoS routing. Master's thesis, North Carolina State University, Raleigh, NC, August 2005. (2006 Graduate School Nancy G. Pollock MS Thesis Award).

[2] D. Cavendish, K. Murakarni, S-H. Yun, O. Matsuda, and M. Nishihara. New transport services for next-generation SONET/SDH systems. *IEEE Communications Magazine*, 40(5):80–87, May 2002.

[3] L. E. Jackson and G. N. Rouskas. Optimal granularity of MPLS tunnels. In *Proceedings of the Eighteenth International Teletraffic Congress (ITC 18)*, pages 1–10. Elsevier Science, September 2003.

[4] L. E. Jackson, G. N. Rouskas, and M. F. M. Stallmann. The directional p-median problem: Definition, complexity, and algorithms. *European Journal of Operations Research*, 2007. (In Press).

[5] C-T. Lea and A. Alyatama. Bandwidth quantization and states reduction in the broadband ISDN. *IEEE/ACM Transactions on Networking*, 3(3):352–360, June 1995.

[6] S. Ramabhadran and J. Pasquale. Stratified round robin: A low complexity packet scheduler with bandwidth fairness and bounded delay. In *Proceedings of ACM SIGCOMM '03*, pages 239–249, August 2003.