A Load Profiling Approach to Routing Guaranteed Bandwidth Flows*

IBRAHIM MATTA
College of Computer Science
Northeastern University
Boston, MA 02115
matta@ccs.neu.edu

AZER BESTAVROS
Computer Science Department
Boston University
Boston, MA 02215
best@cs.bu.edu

Abstract

We study a new approach to routing multi-class traffic flows with guaranteed bandwidth requirements. The approach is based on our recently proposed concept of load profiling *[4]. We thoroughly characterize routing performance using load profiling and contrast it to routing using load balancing and load packing. We do so both analytically and via extensive simulations on Virtual Path (VP) based networks. Our findings confirm that load balancing is not desirable as it results in VP bandwidth fragmentation, which adversely affects the likelihood of accepting new flow requests. This fragmentation is more pronounced when the granularity of the requests is large. Our simulation results also show that our load-profiling routing scheme performs better or as well as the traditional load-balancing routing in terms of revenue under both skewed and uniform workloads. Furthermore, load-profiling routing improves routing fairness by proactively increasing the chances of admitting high-bandwidth flows.*

1 Introduction

Routing algorithms—allowing the selection of one out of many candidate source-to-destination paths for bandwidth reservation purposes—play a critical role in meeting the stringent Quality of Service (QoS) requirements of real-time applications over high-speed integrated services networks, such as Asynchronous Transfer Mode (ATM) networks and next generation Internet.

To support real-time QoS we adopt the *Virtual Circuit* (VC) model for resource reservation. Under this model, routing a connection (or VC) involves the selection of a path (or *route*) within the network from the source to the destination in such a way that the resources (*e.g.*, *bandwidth*) necessary to support the VC QoS requirements are set aside (or *reserved*) for use by the entity requesting the establishment of the VC. This entity might be an application or a router/switch. In the latter case, a router may request a VC to another router to carry the packets of a particular class of applications over a backbone network that connects internet service providers and supports VC routing through IP switching [11] or similar schemes such as tag switching [13], ARIS [5], etc. Over the last few years, several routing protocols based on the VC model have been proposed (*e.g.*, [2, 12]).

We consider a network that supports $S \geq 2$ classes of VCs. A VC of class s requires the reservation of a certain amount of bandwidth b_s that is enough to ensure a given QoS. This bandwidth can be thought of either as the peak transmission rate of the VC or its "effective bandwidth" [6] which varies between the peak and average transmission rates. Without loss of generality, we assume that the bandwidths requested by different classes are distinct and that the classes are indexed in increasing order of their requested bandwidths, i.e., $b_1 < b_2 < \cdots < b_S$.

To support a class-s VC, the VC has to be setup on some path from the source to the destination; the QoS demand (b_s) is allocated on one of the candidate paths for the lifetime of the VC. The objective of the routing algorithm is to choose routes that result in high successful VC setup rate (or equivalently, high carried VC load) while maximizing the utilization of network resources (or equivalently, revenue).

Related Work

Traditionally, routing schemes have been based on the least-loaded concept (*e.g.*, [1, 3]). According to this concept, a request is serviced by setting up the VC on the least utilized path selected from the set of candidate paths[1] between the source and destination, provided it can support the VC's bandwidth requirement. Thus, this scheme attempts to evenly distribute the load among the candidate routes. We call such scheme

*This work was supported in part by NSF CAREER grant NCR-9701988, NEU grant RSDF-377090, and NSF grant CCR-9706685.

[1] To consume the least amount of resources, the set of candidate paths is typically chosen from the set of *shortest* paths.

Least Loaded Routing (LLR).

As an alternative to the load-balancing philosophy of LLR techniques, VC *packing* techniques were proposed in [7]. The argument for VC packing is based on the observation that in order to maximize the utilization of available resources, a routing policy in a heterogeneous (multi-rate) environment should implement *packing* of narrowband VCs (having relatively small bandwidth requirement) on some paths in order to leave room on other paths for wideband VCs (having relatively large bandwidth requirement). Packing strategies achieve two desired properties: (1) They minimize the fragmentation of available bandwidth, resulting in an (2) improved fairness by increasing the chances of admittance for wideband VCs.

A routing scheme based on this packing concept was proposed in [7]. The scheme attempts to pack class-s VCs in order to reduce blocking only for the next higher class of VCs. In [8], we extended the scheme in order to reduce blocking for *all* higher classes. Both schemes are, however, based on pessimistic/deterministic analysis. They only account for the different bandwidth requirements of different classes, but not on their traffic intensities (demands). These traffic intensities may be known a priori (based on traffic forecasts) or dynamically estimated as is often done in telephone networks [3].

In [4], we have established the inadequacy of load-balancing techniques and the impracticality of load-packing techniques. We have done so by analytically characterizing LLR and a load packing heuristic referred to as *Most Loaded Routing* (MLR). MLR selects from the set of candidate routes the most utilized one provided it can support the VC. MLR is a simple scheme which attempts to achieve the same effect as packing-based schemes. We have shown that MLR is asymptotically optimal. We have also introduced in [4] an attractive alternative to MLR that is based on *load profiling*. Our load-profiling VC routing scheme is based on the probabilistic selection of routes, where probabilities are chosen to match the distribution of traffic demand of different classes (*i.e.* the load profile) with the distribution of available resources on the candidate routes (*i.e.* resource availability profile). We call this scheme *Load Profiling Routing* (LPR). In [4], we conducted a pilot simulation experiment that compared LPR to MLR and LLR using a simplified model of a single source-destination node pair connected by multiple paths, where the cost of a path is defined by its current available bandwidth. Our results show that MLR and LPR are competitive and that they both significantly outperform LLR.

Contributions

In this paper we thoroughly characterize the performance of VC routing using load profiling and contrast it to routing using load balancing and load packing. We do so both analytically and via extensive simulations. We extend the analysis in [4] to emphasize the effect of VC request granularity. We then present a much more detailed simulation study that pits LPR to LLR in a more realistic networking environment. In particular, we consider a fully-connected virtual path based network, where routing algorithms consider one-link and two-link paths. Here, the cost of a path is defined by not only its current available bandwidth but also its length; establishing a VC on a two-link path consumes twice as much bandwidth as a one-link path.

Our findings confirm that for reservation-based protocols—which allow for the exclusive use of a preset fraction of a resource's bandwidth for an extended period of time—load balancing is not desirable as it results in resource fragmentation, which adversely affects the likelihood of accepting new reservations. This fragmentation is more pronounced when the granularity of VC requests is large. Typically, this occurs when a common VC is established to carry the *aggregate* traffic flow of many high-bandwidth real-time sources. For virtual path based networks, our simulation results show that our load-profiling VC routing scheme performs better or as well as the traditional load-balancing VC routing in terms of revenue under both skewed and uniform workloads. Furthermore, load-profiling routing improves routing fairness by proactively increasing the chances of admitting high-bandwidth connections. These results support our preliminary investigation in [4] and indicate that LPR is a promising routing approach. We do not show simulation results for MLR and other packing-based schemes since we also found LPR to provide better or similar performance.

The remainder of this paper is organized as follows. Section 2 motivates load profiling by comparing it to load balancing and load packing. In Section 3 a comprehensive comparative evaluation of LPR versus LLR on a fully-connected virtual path based network is presented. We conclude in Section 4 with a summary and with directions for future work.

2 Load Profiling: On Neither Balancing nor Packing VC Requests

2.1 MLR vs LLR: An analytical comparison

Consider a system with M different paths between a particular source and a particular destination. Without loss of generality, we assume that the capacity of all such paths is identical and is normalized to a unit. Let $f(u)$ denote the probability density function for the utilization requirement of requests for VCs between the same source and destination considered above. That is $f(u)$ is the probability that the bandwidth requirement of a VC request will be u, where $0 \leq u \leq U$, where U is the largest possible bandwidth request. By virtue of the capacity assumption, $U \leq 1$.

Let W denote the overall load of the system, expressed as the sum of the reserved bandwidth over all paths (i.e. $0 \leq W \leq M$). A load-balanced system would tend to distribute its load (i.e. reserved bandwidth) equally amongst all paths, making the reserved bandwidth on each path as close as possible to W/M. A load-profiled system would tend to distribute its load in such a way that the probability of satisfying the QoS requirements of incoming VC requests is maximized. We explain below a particular way of achieving such a goal.

Let $\mathcal{C}$ denote the set of M paths in the system. For routing purposes, we assume the availability of a *routing policy* that allows the routing protocol to select a subset of routes from $\mathcal{C}$ that are *believed* to be capable of satisfying the QoS requirement u of an incoming VC request. We denote this *feasible set* by $\mathcal{F} \subseteq \mathcal{C}$.

Let $l_{\mathcal{F}}(u)$ denote the fraction of paths in a feasible set $\mathcal{F}$, whose *unused* (i.e. unreserved/available) bandwidth is equal to u. Thus, $L_{\mathcal{F}}(u) = \int_0^u l_{\mathcal{F}}(u)du$ could be thought of as the (cumulative) probability that the available bandwidth for a path selected at random from $\mathcal{F}$ will be less than or equal to u. Alternatively, $1 - L_{\mathcal{F}}(u)$ is the cumulative probability that the available bandwidth for a path selected at random from $\mathcal{F}$ will be larger than or equal to u, and thus enough to satisfy the demand of a VC request of u (or more) bandwidth.

Thus, the probability that a VC request will be accepted on a path selected randomly out of $\mathcal{F}$ is given by:[2]

$$P = \int_0^U f(u)(1 - L_{\mathcal{F}}(u))du \qquad (1)$$

Let $l_{\mathcal{C}}(u)$ denote the fraction of paths in the system candidate set $\mathcal{C}$, whose unused bandwidth is equal to u. Denote by $L_{\mathcal{C}}(u)$ the cumulative distribution of available bandwidth for $\mathcal{C}$, i.e. $L_{\mathcal{C}}(u) = \int_0^u l_{\mathcal{C}}(u)du$.

Load Balancing: In a perfectly load-balanced system, any feasible set of routes will be identical in terms of its bandwidth profile to the set of all routes in the system. Thus, in a load-balanced system $L_{\mathcal{F}}(u) = L_{\mathcal{C}}(u) = L(u)$. Moreover, we have:

$$L(u) = \begin{cases} 0 & \text{if } 0 \leq u < (1 - W/M) \\ 1 & \text{if } (1 - W/M) \leq u \leq 1 \end{cases} \qquad (2)$$

The probability that a VC request will be accepted is given by $P = \int_0^V f(u)\ 1\ du$, where $V = \min(U, (1 - W/M))$. Thus,

$$P = \begin{cases} F(1 - W/M) & \text{if } 1 < \frac{U}{1-W/M} \\ 1 & \text{if } \frac{U}{1-W/M} \leq 1 \end{cases} \qquad (3)$$

[2] The integration is from 0 to U since U is the largest possible bandwidth request, *i.e.* $f(u) = 0$ for $U < u \leq 1$.

Equation (3) indicates that the performance of LLR is *dependent* on the system load. In particular, equation (3) predicts that LLR's performance will be optimal as long as the utilization of the system (W/M) is less than $1 - U$, but that it will degenerate as soon as (W/M) bypasses that bound. The manner in which such a degeneration occurs will depend heavily on the distribution of requests $f(u)$.

Load Packing: A load-profiling algorithm would attempt to *shape* $L_{\mathcal{C}}(u)$ in such a way that the choice of a feasible set $\mathcal{F}$ would result in minimizing the value of $L_{\mathcal{F}}(u)$, thus maximizing the value of P in equation (1) subject to the boundary constraint $\int_0^1 u\ l_{\mathcal{C}}(u)du = (1 - W/M)$. One solution to this optimization problem is for $l_{\mathcal{C}}(u)$ to be chosen as $l_{\mathcal{C}}(u) = (W/M).\delta_u(0) + (1 - W/M).\delta_u(1)$ where $v.\delta_u(x)$ is an impulse function of magnitude v applied at $u = x$.

The above solution corresponds to a system that *packs* its load (or reserved bandwidth) using the minimal possible number of routes. In other words, a fraction W/M of the paths in the system are 100% utilized, and thus have *no* extra bandwidth to spare, whereas a fraction $(1-W/M)$ of the paths in the system are 100% idle, and thus able to service VC requests with *any* QoS requirements. The choice of any feasible set $\mathcal{F}$ from the set of unused routes in $\mathcal{C}$ would result in $L_{\mathcal{F}}(u)$ being a step function given by:

$$L_{\mathcal{F}}(u) = \begin{cases} 0 & \text{if } 0 \leq u < 1 \\ 1 & \text{if } u = 1 \end{cases} \qquad (4)$$

Plugging these values into equation (1), we get

$$\begin{aligned} P &= \int_0^U f(u)\ (1 - 0)\ du \\ &= 1 \end{aligned} \qquad (5)$$

Equation (5) shows that choosing $l_{\mathcal{C}}(u) = (W/M).\delta_u(0) + (1 - W/M).\delta_u(1)$ is obviously optimal. Furthermore, this optimality is *independent* of the system load or the request distribution $f(u)$.

The *perfect fit* implied in equation (4) may require that VCs already in the system be reassigned to a different path upon the submission and acceptance of a new VC request, or the termination of an existing VC. Even if such reassignment is tolerable, achieving a perfect fit is known to be NP-hard. For these reasons, heuristics such as *first-fit* or *best-fit* are usually employed for on-line resource allocation. Asymptotically, both the first-fit and best-fit heuristics are known to be optimal for the on-line *bin packing* problem [10]. However, for a small value of M—which is likely to be the case in network routing problems—best-fit (or an MLR policy) outperforms first-fit.

2.2 MLR vs LLR: Effect of request granularity

An important distinction between LLR and MLR—evident from equations (3) and (5)—is the sensitivity (insensitivity) of LLR (MLR) to the request distribution $f(u)$. LLR's sensitivity to request distributions is pronounced most when the granularity of the requests is large—*i.e.* U approaches 1—and is insignificant when the granularity of the requests is small—*i.e.* U approaches 0.

To demonstrate the susceptibility of LLR, consider a uniform request distribution over the $[0 - 1]$ interval. According to equation (3), only one half of all VC requests will be possible to honor when the system utilization is 50%, and only one tenth when the system utilization is 90%.[3]

2.3 Load Profiling: A robust alternative to MLR

Equation (4) shows analytically that best-fit (or an MLR policy)—as an approximation of a perfect fit—is an appropriate heuristic for selecting a route from amongst a set of routes that satisfy the bandwidth requirement of a VC request. However, the performance of MLR is severely affected by the inaccuracy of knowledge about reserved bandwidth on various routes that is inherent in a networking environment. In particular, with MLR, it becomes more likely that a VC request gets blocked because the bandwidth available on the selected path turns out to be smaller.

In the remainder of this section, we examine the details of a probabilistic load-profiling heuristic (LPR) that is more appropriate for the imprecision often encountered in distributed and networking environments. Using this LPR protocol, the process of choosing a target route from the set of feasible routes is carried out in such a way so as to maximize the probability of admitting future VC requests. The probability of picking a route from the set of feasible routes is adjusted in such a way that the bandwidth availability profile of the system is maintained as close as possible to the expected profile of incoming VC bandwidth requests.

As an alternative to LLR and MLR, LPR's use of probabilistic route selection results in using *multiple* paths simultaneously *during* a routing information update as opposed to using a single path (the least-loaded or most-loaded). This multi-path routing would further improve performance, and allow for using even a longer routing update interval, thus reducing routing (processing and communication) overheads.

We explain our implementation of LPR through an illustrative example. Consider four classes of VCs with bandwidth requirements b_1, b_2, b_3 and b_4. Without loss of generality, assume $b_1 < b_2 < b_3 < b_4$. Assume the arrival rates are λ_1, λ_2, λ_3 and λ_4. Figure 1 shows the corresponding load profile, i.e. the distribution of requested bandwidths—Prob[requested bandwidth $\leq b_s$], $s = 1, 2, 3, 4$. It also shows the bandwidth availability profile, i.e. the frequency of routes with available bandwidth $\leq b_s$.

Smallest route set	Weight of choosing the path
R_1	$d_1 + d_2 + d_3 + d_4$
R_2	$d_2 + d_3 + d_4$
R_3	$d_3 + d_4$
R_4	d_4

Table 1. Weight assigned to various routes.

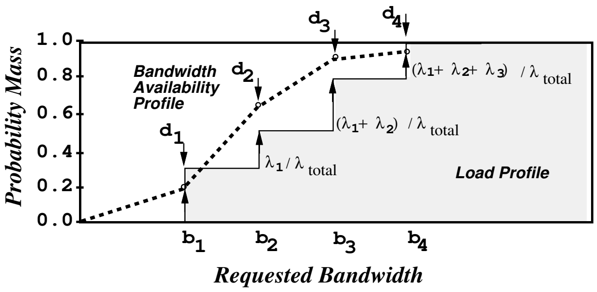

Figure 1. Example load profile and bandwidth availability profile.

The goal of LPR is to make the two profiles match as closely as possible. Denote by R_s the set of paths whose available bandwidth $\leq b_s$. These sets of routes are related as follows: $R_1 \subseteq R_2 \subseteq R_3 \subseteq R_4$. For a new incoming VC, we want to assign it a route from one of these sets. To do so, we compute the probability of choosing a path from each of the route sets. Let d_s be the differences between the load profile and the bandwidth availability profile (see Figure 1). We now assign a weight to each path according to the smallest route set it belongs to as shown in Table 1.[4] To compute a probability distribution, we scale the second column in Table 1 such that all values are non-negative. From the set of feasible paths we select a path probabilistically according to the resulting distribution.

In general, for S classes of VC requests, if R_k is the smallest route set to which a path p belongs, then the weight given to select p, $W(p, k)$, is given by:

$$W(p,k) = \sum_{i=k}^{S} (d_i - d_{min})$$

where $d_{min} = min_s(\{d_s : s = 1, \cdots, S\})$. The com-

[3] For a request distribution with half the granularity–*i.e.* a uniform distribution over the $[0 - 0.5]$ interval—all VC requests will be possible to honor when the system utilization is 50%, and one fifth when the system utilization is 90%.

[4] Note that if a path $p \in R_i$ then $p \in R_j$ for all $j > i$.

plexity of this computation is proportional to the number of VC classes and candidate paths.

3 Simulation of LPR and LLR in Virtual Path Based Networks

In this section, we compare LLR and LPR in a network that uses the Virtual Path (VP) concept [7]. Here a virtual fully-connected network can be overlayed over the physical network, where the VPs constitute the (virtual) links connecting the network nodes. Simple routing schemes that only consider paths with one link (called *direct routes*) and two links (called *alternative routes*) are then used. For a fully-connected network with N nodes, each pair of nodes has one direct route and $N-2$ two-link alternative routes.

3.1 Simulation Model and Setup

We consider a fully-connected logical VP network. We assume all VP links have the same total bandwidth. The network is used by a number of VC classes. A class-s VC requires the reservation of b_s units of bandwidth. We classify bandwidth demands into two categories: 1) *aggregate flow demands*, where the establishment of a VC requires the reservation of a large fraction of the total link bandwidth; and 2) *small flow demands*, where a VC bandwidth requirement is a small fraction of the total link bandwidth. As pointed out earlier, aggregate flow demands could constitute the workload on a multi-class backbone network where a node/router would request the establishment of a high-bandwidth VC to carry a type of real-time traffic coming from an internet service provider or a large number of sources. Class-s VC setup requests arrive to the network according to a Poisson process of rate λ_s. Each class-s VC, once it is successfully setup, has a lifetime of exponential duration with mean $1/\mu_s$.

We consider both uniform and skewed workloads. For a *uniform* workload, the source and destination nodes of an arriving VC are chosen randomly. Each VC class has the same arrival rate and average lifetime. Thus, on average, each node pair has the same VC traffic intensity for each class. In practice, workload is naturally skewed and each node pair may have different VC traffic intensities. To model a *skewed* workload, we assume each VC class has different arrival rate and average lifetime. Furthermore, the network is partitioned into two equal groups, each containing half of the total number of nodes N. The source and destination nodes of a VC are chosen randomly from the same group. The group is chosen with some specified probability, p_{skew}. A node in another group may be chosen by the routing algorithm to act as the intermediate node in a two-link path. We consider routing algorithms that choose from the set of one-link and two-link paths. An arriving VC request rejected by the admission control algorithm—because resources are either unavailable or being reserved for future incoming VCs—is considered blocked and lost.

3.2 Routing Algorithms

Since we are considering routing over paths with different length (in terms of number of links), we have to take into consideration the fact that a VC established over a two-link alternative route consumes twice as much bandwidth compared to when the VC is established over the one-link direct route. The trunk reservation concept [3] is often used to address this issue. Here each link has a Trunk Reservation (TR) value associated with it. A two-link alternative route is said to be *TR-permissible* if, for each of its links, the amount of idle bandwidth available exceeds the corresponding trunk reservation level. The idea is to discourage using two-link routes, and thus reserve some amount of bandwidth for future direct VCs.

Before we present more formally the LLR and LPR algorithms with trunk reservation, we first introduce the following definitions.

Idle Capacity: The *idle capacity* of a link is defined as the amount of link bandwidth that is currently not in use. We define the idle capacity of a route as the minimum idle capacity of all its links.

QOS-permissibility: A route is said to be *QOS-permissible* if it has sufficient idle capacity to carry the VC.

TR-permissibility: For simplicity, we will assume that all links have the same TR value. An alternative route is said to be TR-permissible if *only* when it carries at least one direct VC on one of its links, the idle capacity must be greater than or equal to the reservation threshold [8]. This definition of TR-permissibility requires that switches keep track of the number of direct VCs on outgoing links. This avoids unnecessary reservations for direct VCs when not present.

Allowable Alternative Routes: A two-link alternative route is said to be *allowable* if it is both QOS-permissible and TR-permissible.

3.2.1 Least-Loaded Routing (LLR)

The following steps are executed when a new VC arrives:

1. Set up the VC along the direct route if the direct route is QOS-permissible. Otherwise, go to step 2.

2. If no allowable alternative routes are available, then the VC request is rejected. Otherwise, set up the VC on the allowable alternative route with the largest idle capacity, i.e. the least loaded.

3.2.2 Load Profiling Routing (LPR)

LPR constructs the bandwidth availability profile from the current bandwidth available on the direct and alternative routes between the source and destination. It constructs the desired load profile from the class arrival probabilities of incoming VC requests. The following steps are executed when a new VC arrives:

1. Set up the VC along the direct route if the direct route is QOS-permissible. Otherwise, go to step 2.

2. If no allowable alternative routes are available, then the VC request is rejected. Otherwise, assign selection probabilities to allowable alternative routes according to the difference between the bandwidth availability profile and the desired load profile. Select an allowable alternative route probabilistically to setup the VC.

3.3 Performance Measures

To evaluate the performance of the algorithms, our main measure is *revenue*, which is defined as

$$revenue = \sum_{s=1}^{S} \rho_s \, (1 - B_s) \, b_s$$

where $\rho_s = \frac{\lambda_s}{\mu_s}$, and B_s is the blocking probability of class s. The revenue measure reflects the fact that a commercial network provider's earnings depend not only on the number of VCs admitted, but also on the total amount of VC bandwidth in use. Another measure we considered is *carried load* which is defined as the average number of VCs carried by the network. For lack of space, we do not show results for carried load—see [9]. We found LPR to provide lower carried load than LLR as it tends to accept fewer low-bandwidth VCs and more bandwidth-intensive VCs, thus reducing unfairness.

3.4 Simulation Results for Aggregate Flows

Figures 2 and 3 show results for a 20-node network, i.e., $N = 20$. Each VP link has a total of C units of bandwidth. Here we take $C = 20$. We have four classes of VC with $b_1 = 1.0$, $b_2 = 5.0$, $b_3 = 10.0$ and $b_4 = 15.0$. Trunk reservation is not used in these experiments. Figure 2 shows results for a skewed workload. The arrival rates are $\lambda_1 = 0.4\lambda$, $\lambda_2 = 0.3\lambda$, $\lambda_3 = 0.2\lambda$ and $\lambda_4 = 0.1\lambda$, where λ is the total VC arrival rate. The departure rates are $\mu_1 = 0.004$, $\mu_2 = 0.003$, $\mu_3 = 0.002$ and $\mu_4 = 0.001$. We take $p_{skew} = 0.8$. We observe that LPR outperforms LLR in terms of revenue.

Figure 3 shows results for a uniform workload. The arrival rates are $\lambda_s = 0.25\lambda$ for $s = 1, 2, 3, 4$, where λ is the total VC arrival rate. The departure rates are $\mu_s = 0.002$ for $s = 1, 2, 3, 4$. We observe that LPR still has a higher revenue, although the gain from load profiling is less than that obtained in the skewed workload case. The reason is that this gain is reduced due to the negative effect LPR may have on direct VCs as it tends to load two-link alternative paths nonuniformly and may overload some links resulting in some VCs being alternately routed instead of being directly routed over those (overloaded) links. This leads to increased bandwidth consumption. This effect is more pronounced with MLR which *blindly* overload some links. This may result in more VCs being alternately routed and hence lower revenue.

Figure 4 shows the class blocking probabilities for LPR and LLR under the skewed workload with $\lambda = 1$. LPR reduces the unfairness seen by high-bandwidth (class-4) VCs by reducing their blocking by about 7% at the expense of slight increase in blocking for lower classes.

3.5 Simulation Results for Small Flows

Figures 5 and 6 show results for a network with $N = 20$, $C = 96$, and without trunk reservation. We have four classes of VC with $b_1 = 1.3$, $b_2 = 4.1$, $b_3 = 6.7$ and $b_4 = 9.9$. The class arrival and departure rates are set as in Section 3.4. We also compare the LLR and LPR algorithms to a simple DIRECT routing algorithm that uses *only* direct (one-link) paths.

We observe that LLR performs better than LPR. The gain from load profiling is offset by the loss from overloading some links on alternative routes causing VCs to be alternately routed instead of being directly routed on those (overloaded) links. As pointed out earlier, the gain from load profiling in terms of reduced resource fragmentation is less pronounced with smaller demands. In the skewed workload case, both LLR and LPR are significantly superior to DIRECT (as expected) as they make use of available bandwidth on alternative routes.

However, in the uniform workload case, DIRECT significantly outperforms both LLR and LPR. This is due to the uniformity of the traffic, which implies that all node pairs have, on average, equal VC traffic intensity. Thus, it is more beneficial to minimize using alternative routes whose links are then used by direct VCs, thus conserving network bandwidth. To overcome this drawback of adaptive routing, link reservation thresholds should be used so that an adaptive routing algorithm would converge to direct routing as the load on alternative routes increases.

Routing with Trunk Reservation

Optimal reservation thresholds have often been determined assuming a fixed (known) input traffic pattern (e.g. [14]). For each algorithm, we plot the results corresponding to the reservation threshold that maximizes revenue.

We denote by LLR_res (LPR_res) the LLR (LPR) algorithm with trunk reservation. Figure 7 shows that under skewed workload, LPR_res is competitive to LLR_res. Figure 8 shows that under uniform workload, LPR_res and LLR_res schemes also exhibit similar performance. As expected, DIRECT[5] is not significantly worse than both schemes as is the case under skewed workload. In fact, DIRECT starts to provide similar revenue at high λ, where it is more advantageous to completely avoid using alternative routes.

Although in the case of small flows, the gain from LPR_res due to load profiling is overshadowed by its negative effect on direct VCs resulting in similar revenue as LLR_res, load profiling is still beneficial in reducing unfairness seen by high-bandwidth VCs [9].

4 Conclusion and Future Work

We presented a novel approach to routing guaranteed bandwidth flows in virtual path networks. The approach (LPR) is based on our recently proposed concept of load profiling [4]. Compared to traditional least-loaded-based routing (LLR), it provides better revenue for aggregate VC requests. Also, it reduces unfairness among VC classes by reducing blocking for high-bandwidth classes at the expense of increased blocking for low-bandwidth classes.

Future work remains to further improve LPR routing. One issue we are pursuing is to consider the "length" of the VC request, i.e. the lifetime of the VC. This may be useful in achieving a better "profiling". We are also developing mechanisms for the dynamic control of reservation thresholds. This is of practical interest when the input traffic is time-varying.

References

[1] H. Ahmadi, J. Chen, and R. Guerin. Dynamic Routing and Call Control in High-Speed Integrated Networks. In Proc. *ITC'13*, pages 19–26, Copenhagen, Denmark, June 1991.

[2] C. Alaettinoglu, I. Matta, and A.U. Shankar. A Scalable Virtual Circuit Routing Scheme for ATM Networks. In Proc. *ICCCN '95*, pages 630–637, Las Vegas, Nevada, September 1995.

[3] G. Ash, J. Chen, A. Frey, and B. Huang. Real-time Network Routing in a Dynamic Class-of-Service Network. In Proc. *13th ITC*, Copenhagen, Denmark, 1991.

[4] A. Bestavros and I. Matta. Load Profiling for Efficient Route Selection in Multi-Class Networks. In Proc. *IEEE ICNP*, pages 183–190, Atlanta, Georgia, October 1997.

[5] N. Feldman. ARIS Specification. Internet Draft, March 1997.

[5] Note that with DIRECT, no reservation threshold is used since alternative paths are not used.

[6] R. Guerin, H. Ahmadi, and M. Naghshineh. Equivalent Capacity and its Application to Bandwidth Allocation in High-Speed Networks. *IEEE J. Select. Areas Commun.*, SAC-9(7):968–981, September 1991.

[7] S. Gupta. *Performance Modeling and Management of High-Speed Networks.* PhD thesis, University of Pennsylvania, Department of Systems, 1993.

[8] I. Matta and M. Krunz. Packing and Least-Loaded Based Routing in Multi-Rate Loss Networks. In Proc. *IEEE ICC*, pages 827–831, 1997.

[9] I. Matta and A. Bestavros. Evaluation of a Load Profiling Approach to Routing Guaranteed Bandwidth Flows. Technical Report NU-CCS-97-10, Northeastern University, July 1997.

[10] C. McGeoch and J. Tygar. When are best fit and first fit optimal? In Proc. *1988 SIAM Conference on Discrete Mathematics*, 1988.

[11] P. Newman, T. Lyon, and G. Minshall. Flow Labelled IP: A Connectionless Approach to ATM. In Proc. *IEEE INFOCOM '96*, pages 1251–1260, San Francisco, CA, March 1996.

[12] C. Parris and D. Ferrari. A Dynamic Connection Management Scheme for Guaranteed Performance Services in Packet-Switching Integrated Services Networks. Technical Report TR-93-005, International Computer Science Institute, Berkeley, California, January 1993.

[13] Y. Rekhter et al. Tag Switching Architecture Overview. Internet Draft, September 1996.

[14] S. Sibal and A. DeSimone. Controlling Alternate Routing in General-Mesh Packet Flow Networks. In Proc. *ACM SIGCOMM '94*, pages 168–179, September 1994.

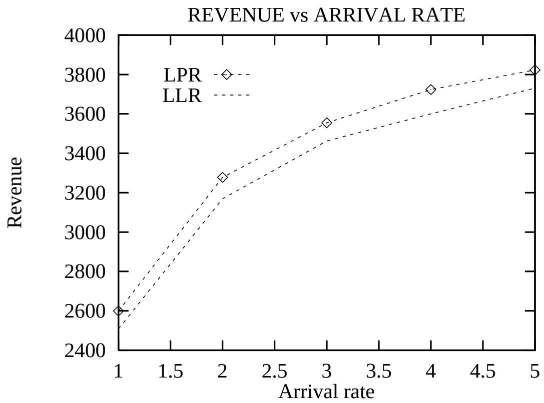

Figure 2. Revenue versus total VC arrival rate. Aggregate flows, skewed workload.

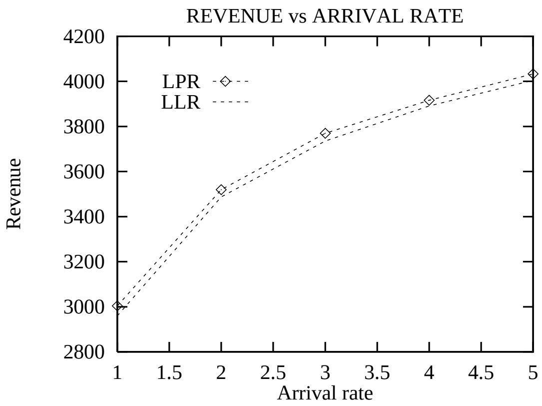

Figure 3. Revenue versus total VC arrival rate. Aggregate flows, uniform workload.

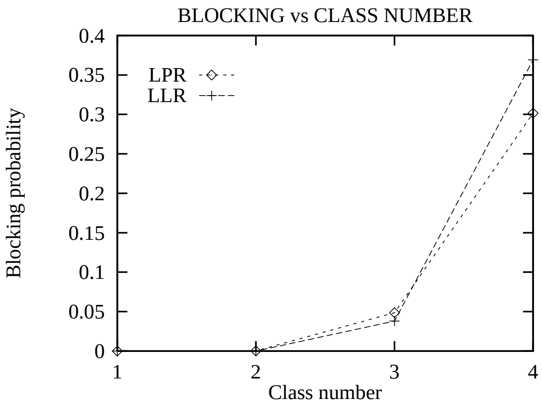

Figure 4. Class blocking probability versus class number. Aggregate flows, skewed workload. VC arrival rate = 1.

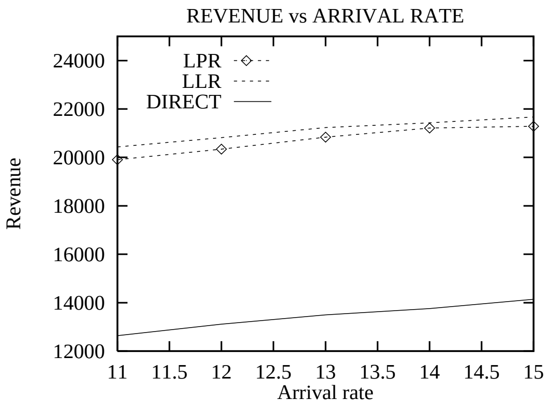

Figure 5. Revenue versus total VC arrival rate. Small flows, skewed workload.

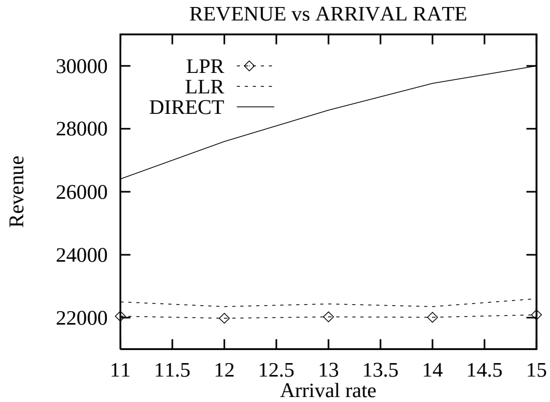

Figure 6. Revenue versus total VC arrival rate. Small flows, uniform workload.

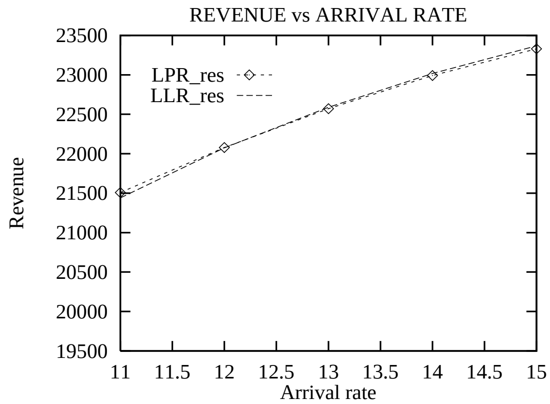

Figure 7. Revenue versus total VC arrival rate. Small flows, skewed workload. With trunk reservation.

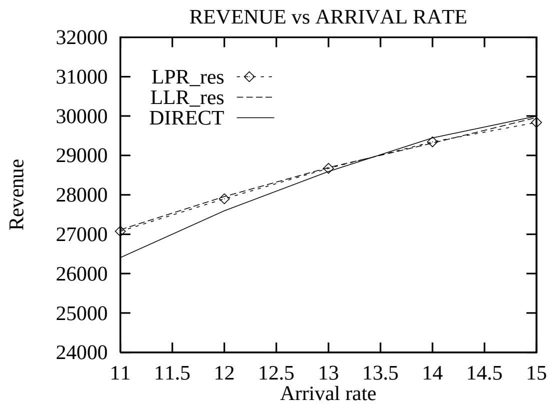

Figure 8. Revenue versus total VC arrival rate. Small flows, uniform workload. With trunk reservation.