A Localized Adaptive Proportioning Approach to QoS Routing

Srihari Nelakuditi and Zhi-Li Zhang, University of Minnesota

ABSTRACT

In QoS routing, paths for flows are selected based on knowledge of resource availability at network nodes and the QoS requirements of flows. Several QoS routing schemes have been proposed that differ in the way they gather information about the network state and select paths based on this information. We broadly categorize these schemes into *best path routing* and *proportional routing*. The best path routing schemes gather *global* network state information and always select the best path for an incoming flow based on this global view. It has been shown that best path routing schemes require frequent exchange of network state, imposing both communication overhead on the network and processing overheads on the core routers. On the other hand, proportional routing schemes proportion incoming flows among a set of candidate paths. We have shown that it is possible to compute near-optimal proportions using only *locally* collected information. Furthermore, a few good candidate paths can be selected using *infrequently* exchanged global information and thus with minimal communication overhead. In this article we describe these schemes in detail and demonstrate that proportional routing schemes can achieve higher throughput with lower overhead than best path routing schemes.

INTRODUCTION

Routing in the current Internet focuses primarily on connectivity and typically supports only "best effort" datagram service. The routing protocols deployed, such as Open Shortest Path First (OSPF), use the *shortest path* routing paradigm, where routing is optimized based on *static* metrics such as hop count or administrative weight. While the service offered by these protocols is suitable for traditional data applications such as ftp and telnet, it is not adequate for many emerging applications such as IP telephony, video on-demand, and teleconferencing, which require stringent delay and bandwidth guarantees. The "shortest paths" chosen for best effort service may not have sufficient resources to provide the requisite service for these applications. Moreover, with explosive growth of Internet traffic, the shortest path routing paradigm of the current Internet also leads to unbalanced traffic distribution: links on frequently used shortest paths become increasingly congested, while links not on shortest paths are underutilized. Hence, it is desirable to devise *adaptive* routing schemes that select paths dynamically based on the requested quality of service (QoS) and the state of the network to provide QoS guarantees for flows while balancing the load across the network.

QoS-based routing has been proposed [1] as a way to address these issues. Under QoS routing, a flow requests a specific QoS and is admitted only if the requested QoS can be guaranteed. Paths for flows are dynamically selected based on knowledge of resource availability (referred to as *QoS state*) at network nodes and the QoS requirements of flows. Upon arrival of a flow the source router first selects,[1] based on its view of the network state, a path that is likely to satisfy the requirements of the flow. It then sends a setup request to reserve the requested bandwidth along the path. This request is accepted and the flow is admitted if sufficient bandwidth is available at all links along the path. Otherwise, the request is rejected, in which case the flow is blocked. The goal of a QoS routing scheme is then to minimize the overall flow blocking probability. A survey of various QoS routing schemes can be found in [2].

In QoS routing, some knowledge regarding the (global) network QoS state is crucial in performing judicious path selection. This knowledge can be obtained, for example, through (periodic) information exchange among routers in a network. Under the *best path* routing approach, each router constructs a global view of the network QoS state by piecing together the QoS state information obtained from other routers, and selects the best path for a flow based on this global view of the network state. Examples of the best path routing approach are various QoS routing schemes [3, 4] based on QoS extensions to the OSPF routing protocol. Best path routing schemes work well when each source node has a reasonably *accurate* view of the network QoS state. However, since network resource availability changes with each flow

---

[1] *Here we assume source routing with bandwidth guarantees.*

0163-6804/02/$17.00 © 2002 IEEE

arrival and departure, maintaining an accurate view of the network QoS state is impractical, due to prohibitive communication and processing overheads entailed by frequent QoS state information exchanges. In the presence of inevitable *inaccurate* information regarding the global network QoS state, best path routing schemes suffer degraded performance.

As a viable alternative to the best path routing approach, we proposed a novel *localized proportional routing* approach to QoS routing [5]. Under this proportional routing approach, instead of (periodically) exchanging information with other routers to obtain a global view of the network QoS state, a source router attempts to *infer* the network QoS state from *locally collected flow statistics* such as flow arrival/departure rates and flow blocking probabilities, and performs adaptive proportioning of flows among a set of *candidate* paths based on this local information. As a result, the localized proportional routing approach avoids the drawbacks of the conventional best path routing approach.

Under a pure localized approach, the candidate path set remains static while their proportions are adjusted dynamically. A network node under a localized approach can judge the quality of paths only by routing some traffic along them. Thus, it is not possible to update the candidate path set based on local information alone. On the other hand, due to changing network conditions, a few good candidate paths cannot be selected statically. Hence we proposed a candidate path selection procedure that dynamically selects a *few good* candidates based on *infrequently* exchanged global information [6]. The inaccuracy in candidate path selection is cushioned by adaptively proportioning traffic among candidates. We demonstrate that our approach of localized adaptive proportional routing using only a few good candidate paths yields higher throughput than best path routing schemes. In addition, this performance gain is achieved with lower overhead.

The following sections describe these schemes in detail. We first discuss best path routing schemes and their shortcomings. We then present proportional routing schemes for selecting candidate paths and assigning proportions to these candidate paths. Finally, we evaluate and compare the performance of best path routing and our proportional routing schemes.

BEST PATH ROUTING

The design of any QoS routing scheme involves addressing two fundamental questions: *how to obtain knowledge of the network state*, and *how to select a path for a flow given this knowledge*. The best path routing schemes gather global network state information through link state updates and select the most feasible (best) path for an incoming flow based on the *current* global view. Most of these schemes exchange *instantaneous* available bandwidth information (i.e., available bandwidth at the time of update) and differ in the way paths are selected based on this information.

Path selection algorithms have to deal with the fundamental trade-off between minimizing resource usage and balancing network load. Resource usage by a flow can be minimized by selecting the shortest path, which may be heavily loaded. The network load can be balanced by choosing the least loaded path, which may be longer and hence consume more resources. There are several path selection algorithms proposed that trade off resource utilization and load distribution differently [3, 4]. They include widest shortest path (*wsp*), shortest widest path (*swp*), and shortest distance path (*sdp*). The *wsp* scheme selects the shortest *feasible* path. A path is considered *feasible* if its bottleneck bandwidth (smallest available bandwidth along the path) is greater than or equal to the requested bandwidth. If more than one shortest feasible path exist, *wsp* chooses the one with the largest bottleneck bandwidth. The *swp* algorithm selects the widest path among all feasible paths. If there are several such paths, the one with minimum hop count is chosen. The *sdp* scheme selects a feasible path with the shortest distance. The distance of a path is defined as the sum of inverses of available bandwidths of its links. Among these, *wsp* is the most popular and well studied algorithm for selecting the best feasible path; hence, we use it as a representative best path routing approach.

Best path routing schemes, such as *wsp*, work well when each source node has a reasonably *accurate* view of the network QoS state. However, as network resource availability changes with each flow arrival and departure, maintaining an accurate view of network QoS state is impractical, due to the prohibitive communication and processing overheads entailed by frequent QoS state information exchange. When the update interval is increased to reduce the overhead, performance degrades rapidly since the QoS state information gets outdated too soon. The main problem with best path routing is that it selects the best path treating stale information as accurate. This leads to the so-called *synchronization* problem: after one QoS state update, many source nodes choose paths with shared links because of their perceived available bandwidth, therefore causing overutilization of these links. After the next QoS state update, the source nodes would avoid the paths with these shared links, resulting in their underutilization. Due to such oscillating behavior, it has been shown that when the update interval is large relative to the flow dynamics, the performance of best path routing schemes degrades significantly [5]. Essentially, exchanging highly varying instantaneous information and selecting the best path based on such information is not a good idea.

As an alternative, we proposed a proportional routing approach where candidate paths are selected based on infrequently exchanged average available bandwidth information and flows are adaptively proportioned among the candidate paths based on locally collected information. By exchanging less variable information that does not get outdated too soon and adaptively proportioning traffic that cushions the impact of inaccuracy in candidate path selection, the proposed proportional routing approach avoids the drawbacks of best path routing.

> *As network resource availability changes with each flow arrival and departure, maintaining an accurate view of network QoS state is impractical, due to the prohibitive communication and processing overheads entailed by frequent QoS state information exchange.*

Given the offered traffic load from a source to a destination, the task of a proportional routing scheme is to determine how to distribute the load among the candidate paths between a source and a destination to minimize the overall blocking probability experienced by the flows.

PROPORTIONAL ROUTING

The proportional routing schemes described here assume that one or multiple explicit-routed paths (label switched paths) are set up, using multiprotocol label switching (MPLS [7]), between each source and destination pair. Flows arriving at a source to a destination are routed along one of these explicit-routed paths (hereafter referred to as *candidate paths*). We assume that flows from a source to a destination arrive randomly with a Poisson distribution, and their holding times are exponentially distributed.[2] Hence, the offered traffic load between a source-destination pair can be measured as the product of average flow arrival rate and holding time. Given the offered traffic load from a source to a destination, the task of a proportional routing scheme is to determine how to distribute the load (i.e., route the flows) among the candidate paths between a source and a destination to minimize the overall blocking probability experienced by the flows. In this section we first discuss how to compute proportions given a set of candidate paths and then describe a procedure for selecting the candidate paths.

GLOBAL OPTIMAL PROPORTIONING

We first consider the scenario where each source node knows all the topology information of the network as well as the offered traffic load between every source-destination pair. With global knowledge of network topology and offered traffic loads, the *optimal* proportions for distributing flows among the paths between each source-destination pair can be computed as described below.

Let R_σ denote the set of candidate paths for routing flows between the pair σ and ν_σ. The global optimal proportioning problem can be formulated as the problem of finding the optimal proportions $\{\alpha_r^*, r \in R_\sigma\}$ such that the overall flow blocking probability in the network is minimized; or, equivalently, finding the optimal proportions such that the total carried traffic in the network, $\sum_\sigma \sum_{r \in R_\sigma} \alpha_r \nu_\sigma (1 - b_r)$ is maximized. Here b_r is the blocking probability on path r when a load of $\nu_r = \alpha_r \nu_\sigma$ is routed through r. This global optimal proportional routing problem is a constrained nonlinear optimization problem and can be solved using an iterative procedure based on the sequential quadratic programming method. We refer to this procedure as optimal proportional routing (opr).

LOCALIZED ADAPTIVE PROPORTIONING

The optimal proportioning procedure described above requires global information about the offered load between each source-destination pair. It is also quite complex and thus time-consuming. As an alternative, we proposed a localized proportional routing approach where each source node collects information about the traffic originating from itself and computes proportions based solely on this local information. Global schemes have to gather systemwide traffic metrics and thus are slower to react to changes. Localized schemes, on the other hand, use only local information and thus can adapt to changes faster. Adaptivity is a key feature of localized schemes that makes them more attractive than global ones.

There are several questions that need to be answered regarding localized proportional routing. What type of information can and should be collected locally? What local objectives should be used in computing proportions so that they have good global effect? We are interested in simple strategies that are easy to implement. One such strategy is *equalization of blocking probabilities* (*ebp*) of candidate paths. This *ebp* strategy requires only path-level information: the amount of offered load and the corresponding blocking probability. This information can easily be collected at a source by keeping track of the number of flows routed along a path and the number of flows blocked along that path.

The objective of *ebp* strategy is to find a set of proportions such that flow blocking probabilities on all the candidate paths are equalized (i.e., $b_r = b_r = \ldots = b_r$). The *ebp* strategy can be implemented using the following procedure to compute new proportions in each iteration. First, the current average blocking probability $\bar{b}$ is computed. Then the proportion of load onto a path is decreased if its current blocking probability is higher than the average and increased if lower than the average. The magnitude of change is determined based on the relative distance of b_i from $\bar{b}$ and some configurable parameters to ensure that change is gradual.

We have evaluated the performance of *ebp* and shown that it yields near-optimal performance when proportions are computed offline using the above iterative procedure [8]. In practice, a source observes the blocking probabilities of candidate paths for an observation interval and recomputes proportions at the end of that period. We have found that the relative performance of a practical *ebp* scheme with regard to *opr* degrades as the number of candidate paths increases. This is because not all paths are good candidates, and a localized scheme such as *ebp* has to route some proportion of traffic to a path to measure its quality. Thus, the performance of practical localized schemes depends critically on the number and choice of candidate paths. Hence, it is desirable to devise a mechanism that supplies a few good candidate paths to a localized proportional routing scheme.

CANDIDATE PATH SELECTION

When identifying a set of candidate paths, an important issue that requires attention is the sharing of links between paths. A set of paths that are good *individually* may not perform as well as expected *collectively*. This is due to the sharing of *bottleneck* links. When two candidate paths of a source-destination pair share a bottleneck link, it may be possible to remove one of the paths and shift all its load to the other path without increasing the blocking probability. Thus, by ensuring that candidate paths of a pair do not share bottleneck links, we can reduce the number of candidate paths without increasing blocking probability. A simple guideline to enforce this could be that the candidate paths of a pair be mutually disjoint [9] (i.e., they do not share *any* links). This is overly restrictive, since even with shared links some paths can cause reduction in blocking if those

---

[2] *The applicability of the proposed schemes does not hinge on this assumption [10].*

links are not congested. What matters is not the sharing itself but the *sharing of bottleneck links*. While the sharing of links among the paths is *static* information independent of traffic, identifying bottleneck links is *dynamic* since the congestion in the network depends on the offered traffic and routing patterns. Therefore, it is essential that candidate paths be *mutually disjoint with regard to bottleneck links*.

A basic question that needs to be addressed by any path selection procedure is, what is a "good" path. In general, a path can be categorized as good if its inclusion in the candidate path set decreases the overall blocking probability considerably. To judge the goodness of paths, we introduce the notion of *width* for a set of paths, which is defined as the maximum flow carriable by paths in the set. The amount of flow carriable by a link is given by its average available bandwidth. So the width of a set of paths can be computed given the average available bandwidth information about each link in the network. This information can be obtained through periodic link state updates. This *globally updated* information is then *locally adjusted* to discount the bandwidth usage by the flows routed between the source-destination pair under consideration. The average amount of load that is successfully routed along a path is added to average available bandwidths of corresponding links. After this adjustment, the average available bandwidth on a link reflects the bandwidth usage on it by all source-destination pairs other than the one under consideration. Note that when two paths share a bottleneck link, the width of two paths together is the same as the width of a single path. This notion of width of a path set essentially accounts for the sharing of links between paths.

Based on the notion of width of a path set, we propose a path selection procedure that adds a new candidate path only if its inclusion increases the width. It deletes an existing candidate path if its exclusion does not decrease the total width. When the number of candidate paths reaches the specified limit, *maxcands*, it replaces a candidate path with another path if this change increases the width. In other words, each modification to the candidate path set either *increases the width* or *decreases the number* of candidate paths. We refer to this scheme as widest disjoint paths (*wdp*). Essentially *wdp* selects *widest paths* that are *mutually disjoint with regard to bottleneck links*.

We now illustrate how a *wdp* scheme selects candidate paths using a simple example. Consider the topology shown in Fig. 1. Suppose that source *s* has to recompute candidate paths to destination *d*. There are five possible paths between *s* and *d*. Let us assume that *s* is currently using paths via 2 → 5 and 3 → 7, and proportioning traffic equally between them. Furthermore, assume that the average amount of load successfully routed between *s* and *d* is 40. Let the average available bandwidths of links received by source *s* through global link state updates be as shown in black. Before recomputing candidate paths, source *s* has to perform local adjustment to discount the bandwidth usage by itself. The source *s* is currently contributing a load of 20 each to paths 2 → 5 and 3 → 7. So the average available bandwidths of links along these paths are correspondingly increased by 20. The new values after local adjustment are shown in blue. Essentially source *s* views the available bandwidth on link 5 → *d* as 30 instead of 10, while other sources view it differently.

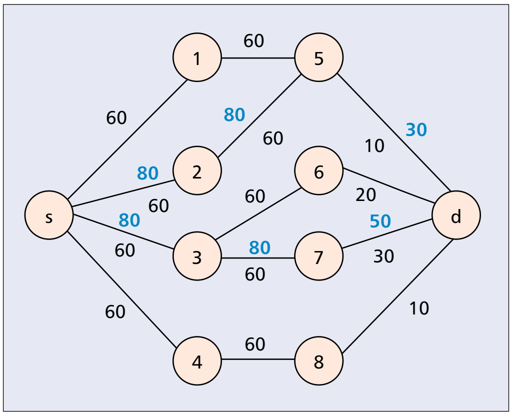

Figure 1. *An illustration of local adjustment.*

Now if the maximum number of candidate paths allowed, *maxcands*, are only two, the candidate path set remains same. This is because the current candidate paths are wider than other paths and replacing any of these paths does not increase the total width. If *maxcands* are more than two, the path via 3 → 6 is added to the candidate set. Only four paths with combined width of 110 would be made candidates even if there is no constraint on the number of candidate paths. The path via 1 → 5 would never be added since it would not increase the total width. Note that although paths *s* → 3 → 6 → *d* and *s* → 3 → 7 → *d* share a link *s* → 3, both are preferred as candidates since the common link is not the bottleneck. On the other hand, *s* → 1 → 5 → *d* is included and *s* → 2 → 5 → *d* is excluded since they share a bottleneck link 5 → *d*. Thus, *wdp* selects widest paths that are mutually disjoint with regard to bottleneck links.

PERFORMANCE EVALUATION

We now evaluate the performance of the proposed schemes. We first describe our simulation setup. Figure 2 shows the topology of an ISP backbone used in our study. All solid links have the same capacity with C_1 units of bandwidth; similarly, all dotted links have C_2 units. Flows arriving into the network are assumed to require one unit of bandwidth. The flow dynamics of the network is modeled as follows. The nodes colored green are considered to be source (ingress) or destination (egress) nodes. Flows arrive at a source node according to a Poisson process with rate λ. The destination node of a flow is chosen randomly from the set of all nodes except the source node. The holding time of a flow is exponentially distributed with mean 1/μ. The offered network load ρ is then given by $\rho = \lambda N \bar{h} / \mu(L_1 C_1 + L_2 C_2)$, where N is the number of source nodes, L_1 and L_2 are the number of solid and dotted links, respectively, and $\bar{h}$ is the mean number of hops per flow averaged across all source-destination pairs. The parame-

Based on the notion of width of a path set, we propose a path selection procedure that adds a new candidate path only if its inclusion increases the width. It deletes an existing candidate path if its exclusion does not decrease the total width.

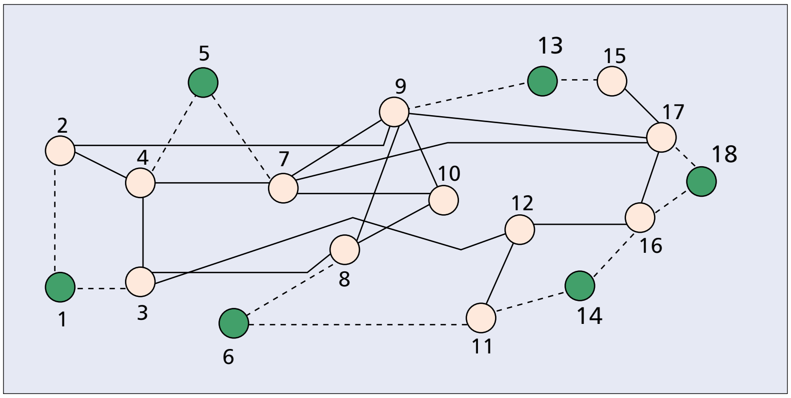

Figure 2. *The* isp *topology used in our study.*

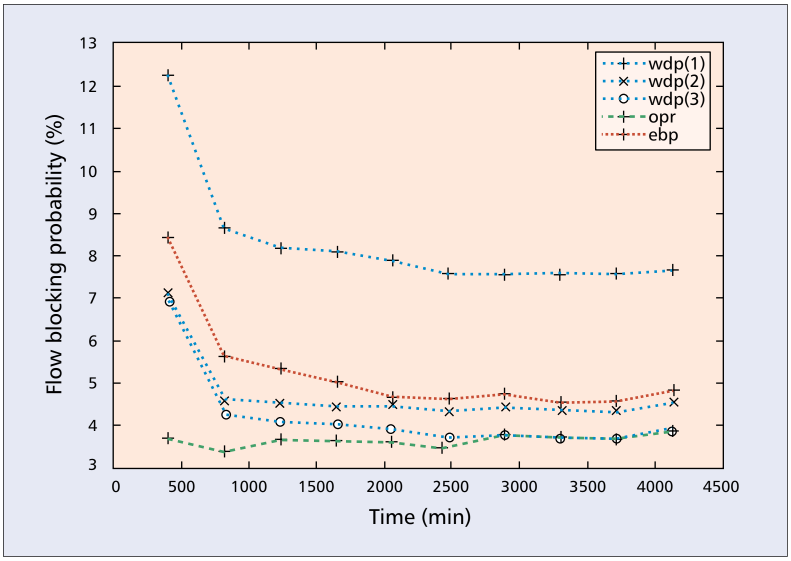

Figure 3. *The convergence process of* wdp.

ters used in our simulations are $C_1 = 20$, $C_2 = 30$, $1/\mu = 1$ min (hereafter written as just m). The topology specific parameters are $N = 6$, $L_1 = 36$, $L_2 = 24$, $\bar{h} = 3.27$. The average arrival rate at a source node λ is set depending on the desired load p.

We now compare the performance of *wdp*, *opr*, and *ebp*. In the case of *opr*, the algorithm is run offline to find the optimal proportions given the set of candidate paths and the offered load between each pair of nodes. All the *minhop* paths and *minhop + 1* paths (i.e., paths that are one hop longer than *minhop*) are chosen as candidates. The resulting proportions are then used in simulation for statically proportioning the traffic among the set of candidate paths. The *ebp* refers to the pure localized scheme where *minhop* paths and *minhop + 1* paths are statically chosen as candidate paths. The *wdp* scheme refers to the proportional routing scheme where widest disjoint paths are chosen as candidates dynamically and traffic is proportioned among them using *ebp*. Ideally, this scheme should be called *wdp + ebp*, but we simply refer to it as *wdp*.

Figure 3 illustrates the convergence process of *wdp* when the load ρ is set to 0.55. It shows overall flow blocking probability as a function of time. The performance of *wdp* is shown for different values of *maxcands*. There are several conclusions that can be drawn from Fig. 3. First, the *wdp* scheme converges for all values of *maxcands*. Second, there is a marked reduction in the blocking probability when *maxcands* is changed from 1 to 2. It is evident that there is quite a significant gain in using multipath routing instead of single-path routing. Third, the *ebp* scheme also converges, albeit slowly. But when *ebp* is used in conjunction with dynamic candidate selection under *wdp*, it converges quickly to lower blocking probability using only a few paths. Finally, using at most three paths per source-destination pair, the *wdp* scheme approaches the performance of the *opr* scheme.

Now let us compare the performance of our proportional routing scheme *wdp* with widest shortest path (*wsp*), a best path routing scheme. Figure 4a shows the blocking probability as a function of update interval used in *wsp*. The update interval for *wdp* is fixed at 30*m*. It is clear that the performance of *wsp* degrades rapidly as the update interval increases. The *wdp* scheme, using at most two paths per pair and infrequent updates with interval of 30*m*, blocks fewer flows than *wsp* that uses many more paths and frequent updates with intervals of 0.5*m*. The performance of *wdp* even with a single path is comparable to *wsp* with intervals of 1.5*m*. Figure 4b displays the flow blocking probability as a function of offered network load. Once again, the update interval for *wdp* is set to 30*m* and the performance of *wsp* is plotted for three different update interval settings: 0.5, 1.0, and 2.0*m*. It can be seen that across all loads the performance of *wdp* with *maxcands* value 2 is better than *wsp* with update interval of 0.5*m*.

There are several factors contributing to the superior performance of *maxcands*. First, the information exchanged about a link is its *average* not *instantaneous* residual bandwidth; hence, less variable and frequent updates are not necessary. Second, the traffic is adaptively proportioned among a few “good” paths instead of loading the “best” path based on inaccurate information. This adaptive proportioning cushions the impact of inaccuracy in candidate selection. Third, globally updated link state is locally adjusted. This makes the network appear different to each source as if they receive customized updates. This prevents the synchronization problem. Moreover, sources using a link perceive more bandwidth on that link than other sources. Consequently, a source continues to use the same set of paths unless other paths are much better. This makes the network more stable. Essentially the nature of information exchanged and the manner in which it is utilized work in a mutually beneficial fashion and lead the system toward a stable optimal state.

CONCLUSIONS

In this article we discuss two broad categories of QoS routing schemes: *best path routing* and *proportional routing*. While best path routing schemes select the best path for each incoming flow, proportional routing schemes proportion flows among a set of candidate paths. We pro-

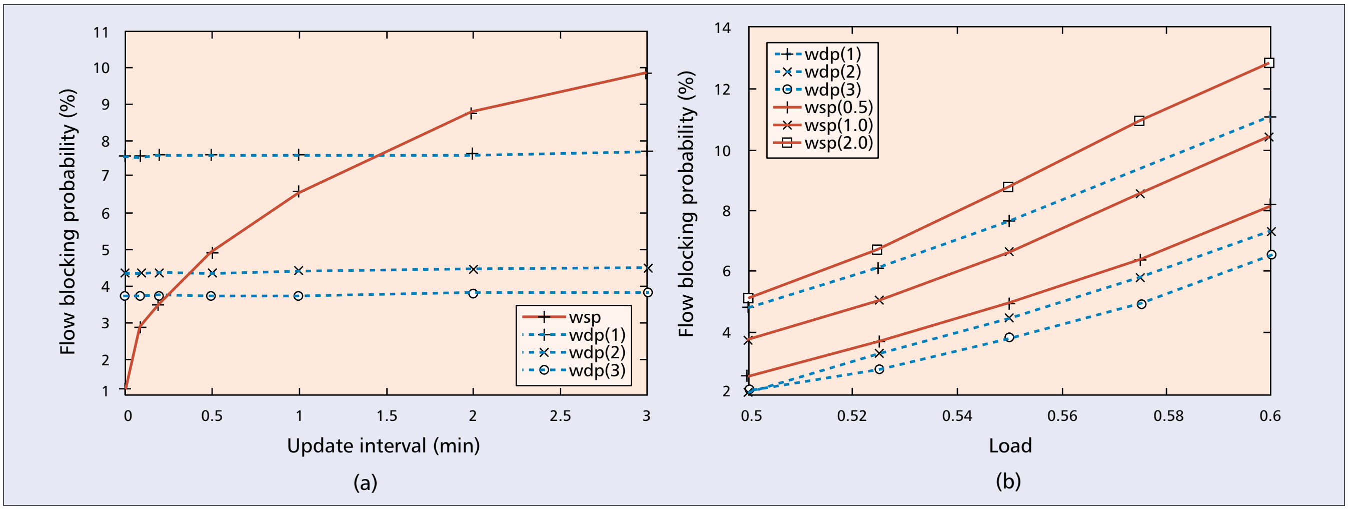

Figure 4. *Performance comparison of* wdp *and* wsp*: a) varying update interval; b) varying load.*

pose a proportional routing scheme that selects *widest disjoint paths* as candidates and proportions flows among them using a simple localized *equalization of blocking probabilities* strategy. We show that our proportional routing scheme yields higher throughput with lower overhead than best path routing schemes. A similar multipath routing approach was used in [10] also. However, our scheme makes routing decisions at the flow level, and consequently the objectives and procedures are different. State-dependent and proportional routing have been studied [11–13, references therein] in the context of telephone networks. These schemes are mainly designed for fully connected networks and not well suited for the Internet. A comparison of some of these schemes with our schemes can be found in [8]. There, we show that our schemes can be used as is to route heterogeneous traffic and also describe extensions for state aggregation and hierarchical routing.

REFERENCES

[1] E. Crawley *et al.*, "A Framework for QoS-Based Routing in the Internet," RFC 2386, Aug., 1998.

[2] S. Chen and K. Nahrstedt, "An Overview of Quality-of-Service Routing for the Next Generation High-Speed Networks: Problems and Solutions," *IEEE Network*, Special Issue on Transmission and Distribution of Digital Video, 1998.

[3] G. Apostolo, "Quality of Service Based Routing: A Performance Perspective," *ACM SIGCOMM '98*.

[4] M. Q. Ma and P. Steenkiste, "On Path Selection for Traffic with Bandwidth Guarantees," *IEEE ICNP '97*.

[5] S. Nelakuditi, Z.-L. Zhang, and R.P. Tsang, "Adaptive Proportional Routing: A Localized QoS Routing Approach," *IEEE INFOCOM 2000*, Mar. 2000.

[6] S. Nelakuditi and Z.-L. Zhang, "On Selection of Paths for Multipath Routing," *IWQOS '01*, June 2001.

[7] E. Rosen, A. Viswanathan, and R. Callon, "Multi-Protocol Label Switching Architecture," Internet draft draft-ietf-mpls-arch-06.txt, Aug. 1999, work in progress.

[8] S. Nelakuditi, "Localized Approach to Providing Quality-of-Service," Ph.D. dissertation, Dept. of Comp. Sci., Univ. of Minn., Dec. 2001.

[9] N. Taft-Plotkin, B. Bellur, and R. Ogier, "Quality-of-Service Routing Using Maximally Disjoint Paths," IWQOS, June 1999.

[10] C. Villamizar, "MPLS Optimized Multipath (MPLS-OMP)," Internet draft, Feb. 1999.

[11] G. Ash, *Dynamic Routing in Telecommunications Networks*, McGraw-Hill, 1998.

[12] P. B. Key and G. A. Cope, "Distributed Dynamic Routing Schemes," *IEEE Commun. Mag.*, Oct. 1990, pp. 54–64.

[13] K. R. Krishnan, "Markov Decision Algorithms for Dynamic Routing," *IEEE Commun. Mag.*, Oct. 1990, pp. 66–68.

BIOGRAPHIES

SRIHARI NELAKUDITI (srihari@cs.umn.edu) received his B.E. in computer science from Andhra University, Visakhapatnam, India, M.Tech. in computer science from the Indian Institute of Technology, Chennai, India, and Ph.D. in computer science from the University of Minnesota, Minneapolis. He is currently a research fellow and an adjunct faculty member in the Department of Computer Science at the University of Minnesota, Minneapolis. His research interests include QoS-based routing, interdomain routing, and video streaming.

ZHI-LI ZHANG (zhzhang@cs.umn.edu) received his B.S. degree in computer science from Nanjing University, China, in 1986, and his M.S. and Ph.D. degrees in computer science from the University of Massachusetts in 1992 and 1997. He is currently an associate professor in the Department of Computer Science and Engineering at the University of Minnesota. He received the NSF CAREER Award in 1997 and the prestigious McKnight Land-Grant Professorship at the University of Minnesota. He currently serves on the Editorial Boards of *IEEE/ACM Transactions on Networking* and *Computer Networks*.