ELSEVIER

Computer Networks 34 (2000) 379–397

COMPUTER NETWORKS

www.elsevier.com/locate/comnet

Hierarchical source routing using implied costs

Michael Montgomery [a,*], Gustavo de Veciana [b,1]

[a] *Center for Information Infrastructure Technology, DOE Y-12 Advanced Technology Directorate, 1099 Commerce Park, Oak Ridge, TN 37830, USA*

[b] *Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX 78712-1084, USA*

Received 21 October 1999; received in revised form 9 February 2000; accepted 14 April 2000

Abstract

Based on a loss network model, we present an adaptive source routing scheme for a large, hierarchically organized network. To represent the "available" capacity of a peer group (subnetwork), we compute the average implied cost to go through or into the peer group. Such implied costs reflect the congestion in the peer group as well as the interdependencies among traffic streams in the network. We prove that both a synchronous and asynchronous distributed computation of the implied costs will converge to a unique solution under a light load condition. Furthermore, we present a more aggressive averaging mechanism that, with sufficient damping, will converge to a unique solution under any traffic conditions. One of the key features of this paper is an attempt to quantify routing "errors" due to inaccuracies caused by aggregation. In fact, our experimental results show that these approximations are reasonably accurate and our scheme is able to appropriately route high level flows while significantly reducing complexity. In addition, we show how on-line measurements and multiservice extensions can be incorporated into the routing algorithm.

Keywords: Hierarchical source routing; Network aggregation; Implied costs; Revenue sensitivities; Adaptive routing

1. Introduction

In order to provide guaranteed quality of service (QoS), communication systems are increasingly drawing on "connection-oriented" techniques. ATM networks are connection-oriented by design, and QoS extensions to the Internet, such as RSVP [6,19,50], make such networks akin to connection-oriented technologies. Indeed, the underlying idea of RSVP is to reserve resources for packet flows, but to do it in a flexible manner using "soft state" which allows flows to be rerouted (or "connections" repacked [23]). Similar comments apply to an IP over ATM switching environment, where IP flows are mapped to ATM virtual circuits. In light of the above trend and the push toward global communication, our focus in this work is on how to make routing effective and manageable in a large-scale, connection-oriented network by using network aggregation. We shall first discuss the importance of using implied costs, introduce hierarchical source routing, explain the basics of our routing algorithm, and give an example of the complexity reduction that it can achieve.

* Corresponding author. Tel.: +1-865-241-1516; fax: +1-865-576-5793; web.: www.ciit.y12.doe.gov/~mcm.

E-mail addresses: montgomerymc@y12.doe.gov (M. Montgomery), gustavo@ece.utexas.edu (G. de Veciana).

[1] Web.: www.ece.utexas.edu/~gustavo.

1389-1286/00/$ - see front matter
PII: S1389-1286(00)00119-5

1.1. Network vs. user optimal routing

In a large-scale network, there are typically multiple paths connecting a given source/destination pair, and it is the job of the routing algorithm to split the demand among the available paths. The routing algorithm which we introduce in this paper fits into the ATM private network-network interface (PNNI) framework [2], or it could replace the border gateway protocol (BGP) [19] in the Internet and split flows in "IP/RSVP" routing. Central to our algorithm is the *implied cost* [22] for a connection along a given path which measures the opportunity cost or expected loss of revenue resulting from accepting a connection. Using implied costs takes into account the possibility of "knock-on" effects (due to blocking and subsequent alternate routing) [22] and is geared towards achieving a *network optimal* routing algorithm.

By routing packets or connections individually so as to minimize their own delays, one may obtain an equilibrium which is *user optimal*. However, since such equilibria are derived from a greedy, somewhat myopic user perspective of the network, they do not usually achieve the minimum *overall* network delays that one would associate with the system optimum from the network provider's point of view. This has been shown to occur in transportation, queueing, and loss networks, as well as other types of networks [3,7,24]. To achieve network optimal routing, one needs to incorporate implied costs into the routing algorithm. The basic idea is that the implied costs correspond to the Lagrange multipliers associated with a network revenue optimization problem. These costs are in turn used to compute the sensitivity of the network revenue to placing additional loads and/or shifting loads on candidate routes in the network, and interdependencies among traffic streams are taken into account.

1.2. Hierarchical source routing: motivation and example

Source routing, where the source specifies the entire path for a connection, is an attractive routing method for connection-oriented networks because a path that provides acceptable QoS and increases network revenue can be chosen up front. By contrast, with hop-by-hop routing, each switch needs to evaluate the QoS across the entire network to determine the next hop [1]. Source routing has the additional advantage that there is no need to run a standardized routing algorithm to avoid loops and policy issues such as provider selection are easily accommodated. For source routing to be effective, we must maintain at least a rough global view of the network state at each host. Propagating information for each link throughout the network quickly becomes unmanageable as the size of the network increases, so a hierarchical structure, such as that proposed in the ATM PNNI specification [2], is needed. Groups of switches are organized into *peer groups* (also referred to as *clouds*), and peer group leaders are chosen to coordinate the representation of each group's state. These collections of switches then form peer groups at the next level of the hierarchy and so on. Nodes keep detailed information for elements within their peer group. For other peer groups, they only have an approximate view of the current state, and this view can become coarser as the "distance" to remote areas of the network increases. We refer to the formation of peer groups as *network aggregation*. Besides reducing the amount of exchanged information, a hierarchical structure permits the use of different routing schemes at different levels of the hierarchy.

By combining a hierarchical network with (loose [2]) source routing, we have a form of routing referred to as *hierarchical source routing*. As an illustration, Fig. 1 shows a fragment of a larger network (Network 0) in which Peer Group 2 contains Nodes 1, 2, and 3. [3] These nodes contain 3, 5, and 4 switches, respectively. To specify, for example, the source at Switch 2 of Node 1 of Peer Group 2 in Network 0, we use the 4-tuple 0.2.1.2. The example in Fig. 1 shows a source at 0.2.1.2 and destination at 0.2.3.4. The source 0.2.1.2 has

[2] In *loose* source routing, only the high-level path is specified by the source. The detailed path through a remote peer group is determined by a border switch of that peer group.

[3] These nodes are peer groups in their own right, but we use the term "node" here to avoid confusion with the peer groups at the next level of the hierarchy.

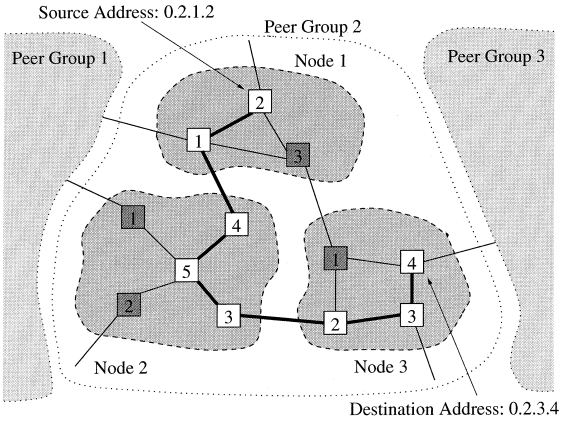


Fig. 1. Illustration of hierarchical addressing and source routing.

specific information about its peer switches 0.2.1.1 and 0.2.1.3, but only aggregated information about nodes 0.2.2 and 0.2.3. The result of performing source routing is a tentative hierarchical path to reach the destination, e.g., $0.2.1.2 \rightarrow 0.2.1.1 \rightarrow 0.2.2 \rightarrow 0.2.3$ which specifies the exact path locally ($0.2.1.2 \rightarrow 0.2.1.1$) then the sequence of remote nodes to reach the destination ($\rightarrow 0.2.2 \rightarrow 0.2.3$). Upon initiating the connection request, the specified path is fleshed out, and, if successful, a (virtual circuit) connection satisfying prespecified end-to-end QoS requirements is set up. In this case, the border switches 0.2.2.4 and 0.2.3.2 in Nodes 2 and 3, respectively, are responsible for determining the detailed path to follow within their respective group. Furthermore, each switch will have a local connection admission control (CAC) algorithm which it uses to determine whether new connection requests can in fact be admitted without degraded performance. If the attempt fails, *crankback* occurs, and new attempts are made at routing the request. [4]

1.3. Explicit vs. implicit representations of available capacity

To do routing in this hierarchical framework, we must decide how to represent the "available" capacity of a peer group, either explicitly or implicitly. The explicit representation takes the physical topology and state of a peer group and represents it with a logical topology plus a metric denoting available capacity that is associated with each logical link. There may also be other metrics such as the average delay associated with logical links.

Typically, the first step in forming the explicit representation is to find the maximum available bandwidth path between each pair of border nodes, i.e., nodes directly connected to a link that goes outside the peer group. If we then create a logical link between each pair of border nodes and assign it this bandwidth parameter, we have taken the *full-mesh* approach [31]. If we collapse the entire peer group into a single point and advertise only one parameter value (usually the "worst case" parameter), we have taken the *symmetric-point* approach [31]. Most proposed solutions lie somewhere between these two extremes.

In the ATM PNNI specification [2], the baseline representation is a star in which each spoke has the same parameter value associated with it. More complex representations are permitted in which *exceptions* have a different associated parameter value than the default. These exceptions can be a spoke of the star or an additional logical link that connects a pair of border nodes. Another alternative is to start with the full-mesh approach and encode the mesh in a maximum weight spanning tree [31]. External nodes can recover the full-mesh representation from the spanning tree if they desire. Whereas the symmetric star topology approximates the "capacity region" of the peer group by a hyper-cube region, the spanning tree approximates it with a hyper-rectangle. A third approach is to approximate the capacity region with a hyperplane [49]. When coupled with prediction of offered loads, the hyperplane approach has the potential to provide a more accurate picture of the available capacity than the star or the spanning tree.

None of the explicit representations, however, are without problems. For example, the maximum available bandwidth paths between different pairs of border nodes may overlap, causing the advertised capacity to be too optimistic. Another

[4] Our model will ignore crankback.

questionable area is scalability to larger networks with more levels of hierarchy. A more important problem is how the representation couples with routing. Can we really devise an accurate representation that is independent of the choice of routing algorithm? None of the explicit representations address the effect that routing calls along particular hierarchical paths would have on the congestion level both within the peer group and in other parts of the network due to interdependencies among traffic streams from various geographic locations. For this reason, we introduce an implicit representation based on the average implied cost to go through or into a peer group that directly addresses this issue and is an integral part of the adaptive hierarchical source routing algorithm that we propose.

1.4. QoS routing based on implied costs

The average implied cost to traverse or enter a peer group reflects the congestion within the peer group as well as the interdependencies among traffic streams across the entire network. Independent of their use in a routing algorithm, they may be useful to network operators for the purpose of accurately assessing current congestion levels as well as providing information valuable for determining the best location for future capacity upgrades and how much they should be willing to pay for them. A rough motivation behind using the average is that, in a large network with diverse routing, a connection coming into a peer group can be thought of as taking a random path through that group, and hence the expected cost that a call would incur would simply be the average over all transit routes through that group. We will develop two closely related approximations: one in which the computed average implied costs are not used for the local portion of a route, and a more aggressive approximation in which the average implied cost is used locally as well as remotely for transit routes traversing more than one peer group. In this second approach, we can conceptualize a route transiting through a peer group as consuming a fraction of bandwidth on each link in that peer group. The fraction used for a particular link would depend on the proportion of actual transit traffic in that peer group which passes through that link.

In order for our scheme to succeed, we need a hierarchical computation of the implied costs and a complementary routing algorithm to select among various hierarchical paths. The path selection will be done through adaptive (sometimes called quasi-static) routing, i.e., slowly varying how demand is split between transit routes that traverse more than one peer group, with the goal of maximizing the rate of revenue generated by the network. After eliminating routes which do not satisfy the QoS constraints, e.g., end-to-end delay, [5] the demand for transit routes connecting a given source/destination pair can be split based on the revenue sensitivities which are calculated using the implied costs. Within peer groups, we feel that dynamic routing should be used since local routing information would be available.

By using an adaptive algorithm based on implied costs, we take the point of view that first it is of essence to design an algorithm that does the right thing on the "average", or say in terms of orienting the high-level flows in the system toward a desirable steady state. In order to make the routing scheme robust to fluctuations, appropriate actions would need to be taken upon blocking/crankback to ensure good, equitable performance in scenarios with temporary heavy loads.

1.5. Using hierarchy to reduce complexity

We now give an example of the complexity reduction achievable with our algorithm. Consider a network consisting solely of Peer Group 2 in Fig. 1. As will be explained in Section 3, the implied costs are computed via a distributed, iterative computation. At each iteration, the links must exchange their current values. Making the assumption that Nodes 1, 2, and 3 are connected locally using a broadcast medium, this would require 81 messages per iteration if we did not employ averaging. With our algorithm for computing

[5] In our model, *effective bandwidth* [9,25] allocation is used to control queueing delays which translates to a limit on hop counts plus propagation delay in order to satisfy a given delay bound.

the implied costs, only 41 messages per iteration would be needed, a savings of 49%. The memory savings would be commensurate with these numbers, and the computational complexity of the two algorithms is roughly the same. This reduction is significant because, in a large-scale network, the overhead associated with information updates in an algorithm such as PNNI can easily overload the network elements [43].

1.6. Related work

Hierarchical routing has been widely studied and used in both telephone and data networks [8,12,15,20,27,46]. Generally, only simple routing metrics such as hop count have been used to select appropriate paths. With the current trend toward integrated broadband networks, interest in QoS-sensitive routing algorithms has been increasing [33,41,48]. In addition, the desire for large-scale networking has made a combination of the above, hierarchical QoS-sensitive routing algorithms, an important area of study [2,16,17,34,40]. For the specific case of routing in ATM networks, which supports QoS and makes use of hierarchy and is consequently quite complex, a good overview can be found in [1]. As an aside, we note that QoS routing problems such as the constrained shortest path problem are typically NP-complete [13,48].

As part of the research on hierarchical QoS-sensitive routing, the explicit representation of available subnetwork capacity has been studied in detail [2,17,31,32,49]. However, our implicit representation based on implied costs is new. Here we have extended the work of Kelly and others on the computation of implied costs and their use in adaptive routing schemes in single-service and multiservice flat networks [11,22,35,37]. Further information on the accuracy and use of implied costs with dynamic routing can be found in [14,26]. Our proposed routing algorithm lies in the class of network optimal algorithms as it attempts to maximize the rate of revenue for the network instead of greedily trying to individually maximize each user's benefit. Network vs. user optimization and the possible effects on stability in QoS-sensitive routing is an issue worthy of further study, especially in light of recent measurements indicating instabilities in current Internet routing [30]. An earlier version of the material in this paper can be found in [39].

1.7. Paper organization

The rest of this paper is organized as follows. Section 2 explains our model and notation. The theoretical basis of our adaptive routing scheme and its relation to Kelly's work [22] is given in Section 3. An alternative approximation of the implied costs that works under any traffic conditions is developed in Section 4. Section 5 presents some computational results which attempt to quantify routing "errors" due to inaccuracies caused by aggregation. In Section 6, we discuss online measurements of some necessary parameters, and Section 7 briefly outlines extensions to a multiservice environment. Finally, Section 8 concludes with a summary.

2. Model and notation

Our model is that of a loss network serving a *single* type of traffic,[6] i.e., all calls require unit bandwidth, call holding times are independent (of all earlier arrival times and holding times) and identically distributed with unit mean, and blocked calls are lost. The unit bandwidth requirement per call can be considered to be an *effective bandwidth* [9,25] which captures the traffic characteristics. The capacity of each link $j \in \mathcal{J}$ is C_j units, and there are a total of J links in the network. Each link j is an element of a single node $n(j) \in \mathcal{N}$, where an aggregated node n is defined as a collection of links that form a peer group or that connect two peer groups.[7] We define E_{jn} to be an indicator function for the event that link j is an element of node n, and P_{jk} is an indicator function for the event that link j is a peer of link k (i.e., in

[6] Extensions to multiservice networks will be presented in Section 7.

[7] There may be multiple links connecting the border switches of two peer groups. This set of one or more interconnecting links is considered to be a separate aggregated node in our model.

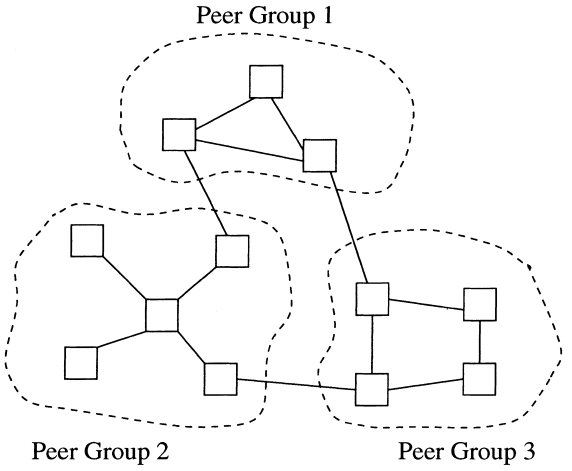


Fig. 2. Example network with a single level of aggregation.

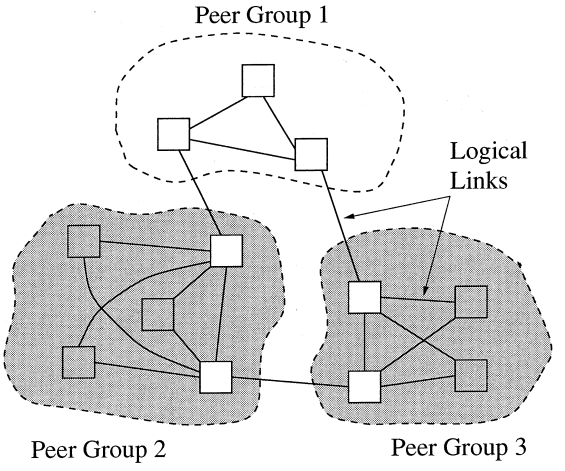


Fig. 3. Logical view of the network from the perspective of peer group 1. The set of links connecting two peer groups is also considered to be an aggregated node in our model.

the same node). A route is considered to be a collection of links in $\mathscr{J}$; route $r \in \mathscr{R}$ uses A_{jr} circuits on link $j \in \mathscr{J}$, where $A_{jr} \in \{0,1\}$.[8] A *transit route* is defined as a route that contains links in more than one node, and T_{nr} is an indicator function for the event that transit route r passes through node n. A call requesting route r is accepted if there are at least A_{jr} circuits available on every link j. If accepted, the call simultaneously holds A_{jr} circuits from link j for the holding time of the call. Otherwise, the call is blocked and lost. Calls requesting route r arrive as an independent Poisson process of rate v_r. Where appropriate, all values referred to in this paper are steady-state quantities.

For simplicity, we only consider a network with one level of aggregation like that shown in Fig. 2. This network has three peer groups, consisting of 3, 5, and 4 switches, respectively. The logical view of the network from a given peer group's perspective consists of complete information for all links within the peer group but only aggregated information for links between peer groups and in other peer groups. The other peer groups conceptually have logical links which connect each pair of border switches and connect each border switch to each internal destination. These logical links have an associated *implied cost*, i.e., marginal cost of using this logical resource, which is approximated from the real link implied costs. Currently, we calculate an average implied cost for any transit route that passes through or into a node, i.e., all of the logical links in a node have the same implied cost, and this value is then advertised to other peer groups. Fig. 3 shows the logical view of the example network from the perspective of peer group 1.

3. Approximations to revenue sensitivity

To calculate the revenue sensitivities, we must first find the blocking probability for each route, an important performance measure in its own right. Steady-state blocking probabilities can be obtained through the invariant distribution of the number of calls in progress on each route. However, the normalization constant for this distribution can be difficult to compute, especially for large networks. Therefore, the blocking probabilities are usually estimated using the Erlang fixed point approximation [15,23].

Let $B = (B_j, j \in \mathscr{J})$ be the solution to the equations

$$B_j = E(\rho_j, C_j), \quad j \in \mathscr{J}, \tag{1}$$

[8] In general, these routes might include multicast routes.

where

$$\rho_j = \sum_{r \in \mathscr{R}} A_{jr} v_r \prod_{k \in r - \{j\}} (1 - B_k) \tag{2}$$

and the function E is the Erlang B formula [4]

$$E(\rho_j, C_j) = \frac{\rho_j^{C_j}}{C_j!} \left[\sum_{n=0}^{C_j} \frac{\rho_j^n}{n!} \right]^{-1}. \tag{3}$$

The vector B is called the Erlang fixed point; its existence follows from the Brouwer fixed point theorem and uniqueness was proved in [21]. Using B, an approximation for the blocking probability on route r, under the assumption that blocking is independent from link to link, is

$$L_r \approx 1 - \prod_{k \in r} (1 - B_k). \tag{4}$$

Alternatively, instead of using the Erlang fixed point to approximate the blocking probabilities, it may be more accurate and efficient to measure the relevant quantities. Specifically, L_r, λ_r (the throughput achieved on route r), and $\theta_j = \sum_{r \in \mathscr{R}} A_{jr} \lambda_r$ (the total throughput through link j) can be obtained based on moving-average estimates. This will in turn allow us to compute the associated implied costs and hence the approximate revenue sensitivities. We will discuss the subject of on-line measurements more fully in Section 6.

Assuming that a call accepted on route r generates an expected revenue w_r, the rate of revenue for the network is

$$W(v; C) = \sum_{r \in \mathscr{R}} w_r \lambda_r. \tag{5}$$

Starting from the Erlang fixed point approximation and by extending the definition of the Erlang B formula (3) to non-integral values of C_j via linear interpolation,[9] the sensitivity of the rate of revenue with respect to the offered loads has been derived by Kelly [22] and is given by

$$\frac{\partial}{\partial v_r} W(v; C) = (1 - L_r) s_r, \tag{6}$$

where

$$s_r = w_r - \sum_{k \in \mathscr{J}} A_{kr} c_k \tag{7}$$

is the surplus value of an additional connection on route r, and the link implied costs are the (unique) solution to the equations

$$c_j = \eta_j (1 - B_j)^{-1} \sum_{r \in \mathscr{R}} A_{jr} \lambda_r (s_r + c_j), \quad j \in \mathscr{J}, \tag{8}$$

where $\eta_j = E(\rho_j, C_j - 1) - E(\rho_j, C_j)$. B_j, ρ_j, and L_r are obtained from the Erlang fixed point approximation, and $\lambda_r = v_r(1 - L_r)$.

In a flat network, the offered load for a given source/destination pair should be split among the available routes based on the revenue sensitivities in (6). An additional call offered to route r will be accepted with probability $1 - L_r$. If accepted, it will generate revenue w_r, but at a cost of c_j for each $j \in r$. The implied costs c quantify the potential knock-on effects or expected loss in revenue due to accepting a call. The goal of the routing algorithm is to maximize the rate of network revenue $W(v; C)$ by adaptively adjusting the splitting for each source/destination pair over time in response to changing traffic conditions. The splitting for a source/destination pair should favor routes for which $(1 - L_r)s_r$ has a positive value since increasing the offered traffic on these routes will increase the rate of revenue. Routes for which $(1 - L_r)s_r$ is negative should be avoided, with all adjustments of the splitting made gradually to guard against sudden congestion. We note that, in general, $W(v; C)$ is not concave, so there may exist nonoptimal local maxima. However, Kelly has shown that it is asymptotically linear when v and C are increased proportionally [22]. Furthermore, even though the routing algorithm could potentially reach a nonoptimal local maximum of the revenue function, the stochastic fluctuations in the offered traffic may allow it to escape that particular region.

To perform aggregation by peer group, we first define the quantity $\bar{c}_n$ as the weighted average of the implied costs associated with pieces of transit routes that pass through or enter node n (or, equivalently, over the links in n visited by such routes) where, in the following, $c_r^n = \sum_{j \in \mathscr{J}} A_{jr} E_{jn} c_j$:

[9] At integer values of C_j, define the derivative of $E(\rho_j, C_j)$ with respect to C_j to be the left derivative.

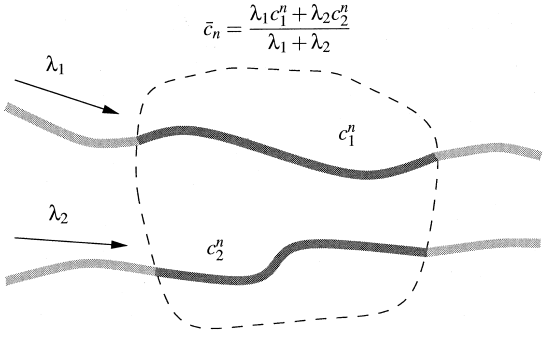


Fig. 4. Computation of $\bar{c}_n$ for an aggregated node n with two transit routes.

$$\bar{c}_n = \frac{\sum_{r\in\mathscr{R}} T_{nr}\lambda_r c_r^n}{\sum_{r\in\mathscr{R}} T_{nr}\lambda_r} = \frac{\sum_{j\in\mathscr{J}} E_{jn}(\sum_{r\in\mathscr{R}} T_{nr}A_{jr}\lambda_r)c_j}{\sum_{r\in\mathscr{R}} T_{nr}\lambda_r}. \quad (9)$$

This averaging is illustrated in Fig. 4. We redefine the surplus value for a route as a function of the local link implied costs and the remote nodal implied costs, *from the perspective of link* $j \in r$ (see Fig. 5)

$$s_{r;j} = w_r - \sum_{k\in\mathscr{J}} A_{kr}P_{kj}c_k - \sum_{n\neq n(j)} T_{nr}\bar{c}_n. \quad (10)$$

The link implied costs are now calculated as

$$c_j = \eta_j(1-B_j)^{-1}\sum_{r\in\mathscr{R}} A_{jr}\lambda_r(s_{r;j}+c_j), \quad j\in\mathscr{J}. \quad (11)$$

In the sequel, we will address the following issues: the existence of a unique solution to these equations, convergence to that solution, and the accuracy relative to the implied costs (8) associated with a flat network.

Eq. (11) can be solved iteratively in a distributed fashion via successive substitution. If we define a linear mapping $f:\mathbb{R}^J\to\mathbb{R}^J$ by $f=(f_1,f_2,\ldots,f_J)$,

$$f_j(x) = \eta_j(1-B_j)^{-1}\sum_{r\in\mathscr{R}} A_{jr}\lambda_r \times\left(w_r - \sum_{k\neq j} A_{kr}P_{kj}x_k - \sum_{n\neq n(j)} T_{nr}\bar{x}_n\right), \quad (12)$$

then successive substitution corresponds to calculating the sequence $f^i(x)$, $i=1,2,\ldots$, where $f^i(x)$ is the result of iterating the linear mapping i times.

Let $\|\cdot\|_M$ denote the following norm on $\mathbb{R}^J$:

$$\|x\|_M = \max_{j,r}\left\{A_{jr}\left(\sum_{k\neq j} A_{kr}P_{kj}|x_k| + \sum_{n\neq n(j)} T_{nr}\overline{|x|}_n\right)\right\}, \quad (13)$$

where

$$\overline{|x|}_n = \frac{\sum_{j\in\mathscr{J}} E_{jn}(\sum_{r\in\mathscr{R}} T_{nr}A_{jr}\lambda_r)|x_j|}{\sum_{r\in\mathscr{R}} T_{nr}\lambda_r}.$$

For any positive vector α, we define the weighted maximum norm on $\mathbb{R}^J$ by $\|x\|_\infty^\alpha = \max_j |x_j/\alpha_j|$, where we suppress the index α if $\alpha_j = 1$ for all j. Also, let $\delta=(\delta_1,\delta_2,\ldots,\delta_J)$, where $\delta_j=\eta_j\rho_j$ denotes Erlang's improvement formula [22].

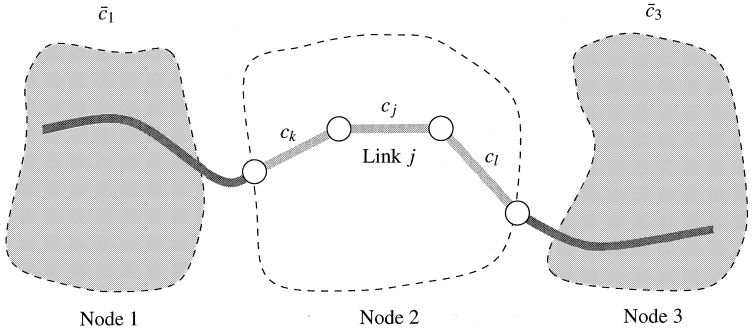


Fig. 5. Implied costs for a route from the perspective of link j.

Theorem 1. *Suppose that* $\|\delta\|_M < 1$. *Then the mapping* $f : \mathbb{R}^J \to \mathbb{R}^J$ *is a contraction mapping under the norm* $\|\cdot\|_M$, *and the sequence* $f^i(x), i = 1, 2, \ldots$, *converges to* c', *the unique solution of* (11), *for any* $x \in \mathbb{R}^J$.

Proof. See Appendix A. □

The product $\eta_j \rho_j$ increases to 1 as ρ_j, the offered load at link j, increases [22]. So $\|\delta\|_M < 1$ can be referred to as a light load condition. If the network has long routes and/or heavily loaded links, this constraint may be violated, but at moderate utilization levels, we expect that it will hold. As an example, consider a loss network in which all links have capacity $C = 150$ and the reduced load at each link from thinned Poisson streams is $\rho = 100$. Furthermore, for simplicity, assume that each transit route across a node has the same length. Then $\delta = 3.3 \times 10^{-5}$ for each link, and the condition $\|\delta\|_M < 1$ requires the maximum route length to be at most 30,717 links. The blocking probability for a route of maximum length is approximately 2% (under the link independence assumption). If ρ is increased to 120 for each link, the maximum route length is 33 links with a blocking probability of approximately 3% along such a route. At $\rho = 140$, the maximum route length is 3 links with a blocking probability of approximately 8%. For this scenario (equal link capacity $C = 150$ and equal loads at each link), link utilizations up to about 80% are certainly feasible under our "light load" condition. As the capacities of the links increase (relative to bandwidth requests), even higher utilizations are possible before the maximum route length becomes too small and/or blocking becomes prohibitive.

The convergence proved in Theorem 1 assumes iterates are computed synchronously. In a large-scale network, synchronous computation may be infeasible, so we will show that our light load condition is sufficient for convergence of an asynchronous computation in the following sense [5]:

Assumption 1 (*Total asynchronism*). Each link performs updates infinitely often, and given any time t_1, there exists a time $t_2 > t_1$ such that for all $t \geqslant t_2$, no component values (link and average implied costs) used in updates occurring at time t were computed before t_1.

Note that, under this assumption, old information is eventually purged from the computation, but the amount of time by which the variables are outdated can become unbounded as t increases.

Theorem 2. *Suppose that* $\|\delta\|_M < 1$ *and* $\delta > 0$. *Then, under Assumption* 1 (*total asynchronism*), *the sequence* $f^i(x), i = 1, 2, \ldots$, *converges to* c', *the unique solution of* (11), *for any* $x \in \mathbb{R}^J$.

Proof. See Appendix B. □

With the additional restriction of bounded communication delays, the convergence rate of an asynchronous iteration satisfying the conditions of Theorem 2 is geometric and can actually be faster than the corresponding synchronous version which has to wait for all values from the previous iteration to be distributed before performing the next update. See [5, pp. 441–443] for the details of a situation analogous to ours which has "fast" local communication (within peer groups) and "slower" remote communication (between peer groups) and where the asynchronous convergence rate is faster if there is a "strong coupling" among the local variables (i.e., the local implied costs), a condition which should typically hold true in a hierarchical network if the amount of local traffic dominates the amount of remote traffic in each peer group.

Theorem 3. *Suppose that* $\|\delta\|_M < 1$ *and denote* c *and* c' *as the solutions to* (8) *and* (11), *respectively. Define* $\Delta = \max_{n,r} \{T_{nr} \sum_{m \neq n} T_{mr} |c_r^m - \bar{c}_m|\}$ *where* $c_r^m = \sum_{j \in \mathcal{J}} A_{jr} E_{jm} c_j$ *and* $\bar{c}_m$ *is defined by* (9). *Then we have*

$$\|s - s'\|_\infty \leqslant \frac{\Delta \|\delta + 1\|_\infty}{1 - \|\delta\|_M}, \tag{14}$$

where by $\|s - s'\|_\infty$ *we mean* $\max_{j,r:j \in r} |s_r - s'_{r;j}|$.

Proof. See Appendix C. □

The error between our modified implied costs (11) and the implied costs (8) associated with a flat network will be minimized under light loads ($\|\delta\|_M \ll 1$) and if the difference between transit route costs and the average for each node is small (Δ close to 0). We use the maximum norm of $s - s'$ as a comparison because it directly affects the difference in the revenue sensitivity in (6) using the flat and hierarchical frameworks. The measured value of L_r used in (6) may also be different from that in a flat network because it is potentially averaged over several routes with the same hierarchical path from a given node's point of view. When making adaptive routing decisions, we are really only concerned with the relative values of $\frac{\partial}{\partial v_r} W(v; C)$ among routes sharing a common source/destination pair. It is unclear in what situations our approximation might affect this ordering.

To summarize, our routing algorithm works as follows:

1. The blocking probabilities and carried loads are first estimated using the Erlang fixed point approximation (or on-line measurements as discussed in Section 6).
2. The link implied costs (11) and average implied costs (9) are computed iteratively by asynchronously exchanging values and recomputing until the costs converge.
3. The implied costs are used to compute the revenue sensitivities $(1 - L_r)s_{r;j}$ for candidate hierarchical paths. (Routes not able to provide the desired QoS, e.g., paths with excessive propagation delays, may have already been eliminated.)
4. The splitting probabilities for the hierarchical paths connecting a source to a given destination peer group are adjusted based on the revenue sensitivities. The higher the sensitivity, the more traffic should be offered to that path, but all adjustments should be made gradually.
5. The process is repeated periodically as conditions warrant.

Underneath this adaptive routing mechanism, a dynamic routing algorithm (which we have not specified) is run within each peer group to route transit and local traffic within that peer group.

4. An alternative approximation

In this section, we consider a more aggressive averaging mechanism. In the previous approach, we used exact information for resources within a peer group and aggregated metrics to represent its remote peers. By contrast, herein we also perform local averaging among routes transiting through or into a local peer group. We will show that this alternative approximation has a similar structure to the previous case, although it is a cruder approximation for the implied costs associated with a flat network. The key advantage of this approach is that, subject to sufficient damping, one can show convergence to new approximate implied costs under any traffic conditions and route topology. In fact, the required damping within a peer group depends only on local information, the number of links within the peer group, and aggregated global information, the total number of peer groups. Thus, the damping factor within a peer group only requires information that is consistent with its hierarchically aggregated view of the network, and the nonlocal knowledge required, namely the total number of peer groups, is not detrimental to the decentralized nature of the computation.

Define the matrix $\bar{A}$ with elements $\bar{A}_{jr} \in [0, 1]$ such that

$$\bar{A}_{jr} = \begin{cases} \dfrac{\sum_{q \in \mathscr{R}} T_{n(j)q} A_{jq} \lambda_q}{\sum_{q \in \mathscr{R}} T_{n(j)q} \lambda_q} & \text{if } T_{n(j)r} = 1, \\ A_{jr} & \text{if } T_{n(j)r} = 0. \end{cases} \tag{15}$$

Local routes remain unchanged: they take a single circuit on each link that they traverse. However, transit routes can be thought of as consuming a fraction of a circuit on every link in each node that they traverse. This fraction is equal to the fraction $\bar{A}_{jr}$ of transit traffic in node $n(j)$ which passes through that link. Note that the offered load ρ_j at link j remains the same whether it is computed based on the flat network's routing matrix A or the aggregated routing matrix $\bar{A}$. Indeed, for fixed λ_r, we have $\rho_j = (1 - B_j)^{-1} \sum_{r \in \mathscr{R}} A_{jr} \lambda_r = (1 - B_j)^{-1} \sum_{r \in \mathscr{R}} \bar{A}_{jr} \lambda_r$.

By substituting $\bar{A}$ for A in (8), we have the following implied cost equations:

$$c_j = \eta_j(1-B_j)^{-1}\sum_{r\in\mathscr{R}}\bar{A}_{jr}\lambda_r\left(w_r - \sum_{k\neq j}\bar{A}_{kr}c_k\right),\quad j\in\mathscr{J}. \tag{16}$$

We can rewrite these equations in various ways to bring out the connections with both our first aggregation method (9) and the original implied cost equation (8) for a flat network. First, we note that for a given link j and route r such that $T_{n(j)r}=1$, we have $\sum_{k\in\mathscr{J}}\bar{A}_{kr}P_{kj}c_k = \bar{c}_{n(j)}$, which illuminates the role of $\bar{A}_{jr}$ in performing additional averaging of implied costs at the local level; compare this with (9). Second, we can rewrite (16) as

$$\begin{aligned} c_j &= \eta_j(1-B_j)^{-1}\sum_{r\in\mathscr{R}}\bar{A}_{jr}\lambda_r \\ &\quad\times\left(w_r - \sum_{k\neq j}\bar{A}_{kr}P_{kj}c_k - \sum_{n\neq n(j)}T_{nr}\bar{c}_n\right) \qquad (17)\\ &= \eta_j(1-B_j)^{-1}\sum_{r\in\mathscr{R}}\Bigg[(1-T_{n(j)r})A_{jr}\lambda_r \\ &\quad\times\left(w_r - \sum_{k\in\mathscr{J}}A_{kr}c_k + c_j\right) + T_{n(j)r}\bar{A}_{jr}\lambda_r \\ &\quad\times\left(w_r - \sum_{n\in\mathscr{N}}T_{nr}\bar{c}_n + \bar{A}_{jr}c_j\right)\Bigg]. \qquad (18)\end{aligned}$$

By comparing (17) with (11), we note that our two approximations differ only in the use of the $\bar{A}$ matrix locally instead of A. In (18), we see that the equation for c_j is a combination of the original Eq. (8) for routes not transiting through node $n(j)$ and an equation based on "averaged" surplus values $s_r = w_r - \sum_{n\in\mathscr{N}}T_{nr}\bar{c}_n$ for routes transiting through node $n(j)$ with $\bar{A}_{jr}$ replacing A_{jr}.

Based on the above, we define a new linear mapping $\tilde{f}:\mathbb{R}^J\to\mathbb{R}^J$ by $\tilde{f}=(\tilde{f}_1,\tilde{f}_2,\ldots,\tilde{f}_J)$,

$$\tilde{f}_j(x) = \eta_j(1-B_j)^{-1}\sum_{r\in\mathscr{R}}\bar{A}_{jr}\lambda_r\left(w_r - \sum_{k\neq j}\bar{A}_{kr}x_k\right), \tag{19}$$

where $\tilde{f}^i(x)$ is the result of iterating the linear mapping i times. Define $\tilde{f}_{(\gamma)}:\mathbb{R}^J\to\mathbb{R}^J$ to be a damped version of the iteration $\tilde{f}(\cdot)$ for $\gamma = \mathrm{diag}(\gamma_j)_j$ where $\gamma_j\in(0,1)\ \forall j\in\mathscr{J}$:

$$\tilde{f}_{(\gamma)}(x) = (I-\gamma)x + \gamma\tilde{f}(x). \tag{20}$$

If we define a norm on $\mathbb{R}^J$ by

$$\|x\|_{\tilde{M}} = \max_{j,r}\left\{1(\bar{A}_{jr}>0)\sum_{k\neq j}\bar{A}_{kr}|x_k|\right\}, \tag{21}$$

then Theorems 1 and 2 can be shown to hold for $\tilde{f}(x)$ under the condition $\|\delta\|_{\tilde{M}}<1$. However, our main interest here lies in proving convergence of the damped iteration $\tilde{f}_{(\gamma)}(x)$ without requiring $\|\delta\|_{\tilde{M}}$ to be less than one.

In the following, let J_n denote the number of links in node n, and recall that N denotes the total number of aggregated nodes in the network.

Theorem 4. *Eqs.* (16) *have a unique solution* $\tilde{c}$. *If* $\gamma_j \leqslant (NJ_{n(j)})^{-1}\ \forall j\in\mathscr{J}$, *then the sequence* $\tilde{f}^i_{(\gamma)}(x)$, $i=1,2,\ldots$, *converges to* $\tilde{c}$ *for any* $x\in\mathbb{R}^J$.

Proof. The proof closely resembles the development in [22, Section 4]. See [38, Section 4.5] for the proof. □

The convergence proved in Theorem 4 is based on synchronous iterations. Our conjecture is that under a partially asynchronous model [5], i.e., there is a fixed bound D on the amount of time by which the information used at a link can become outdated, the algorithm will converge if we use a small enough stepsize γ [38, Section 4.5].

5. Computational results

In this section, we explore the computation of the implied costs at one point in time for a given set of offered loads. We use the Erlang fixed point equations to obtain the route blocking probabilities, and then input the results to the implied cost calculations. Let c, c', and $\tilde{c}$ denote the solutions to (8), (11) and (16), respectively. The surplus values s and s' are computed according to (7) and (10), respectively. For our alternative approximation, we compute $\tilde{s}_r = w_r - \sum_{k\in\mathscr{J}}\bar{A}_{kr}\tilde{c}_k$. Because we use the same route blocking probabilities L in computing the revenue sensitivities for all three cases, the expected and maximum relative surplus value differences are equal to the expected and maximum relative revenue sensitivity errors. The results discussed below are summarized in Tables 1 and 3.

Table 1
Computational results for the four experiments

	Revenue sensitivity error: $\mathbb{E}[\cdot]/\|\cdot\|_\infty$		$\partial W/\partial v_1$	$\partial W/\partial v_2$	Implied cost error: $\mathbb{E}[\cdot]/\|\cdot\|_\infty$		$L_{\max}(\%)$	$\|\delta\|_M$	Iterations
	$(s-s')/s$	$(s-\tilde{s})/s$			$(c-c')/c$	$(c-\tilde{c})/c$			
Symmetric load	0.0%/0.0%	0.01%/0.04%	0.823	0.684	0.0%/0.0%	0.5%/0.7%	2.1	0.297	5
Local overload	0.2%/1.5%	15.0%/97.5%	0.416	0.772	0.1%/0.3%	5.7%/9.0%	25	0.764	10–13
Transit overload	0.9%/5.0%	1.0%/4.4%	0.335	0.686	0.4%/1.1%	6.3%/18.1%	16	0.780	8–9
Asymmetric net	2.4%/15.5%	2.2%/15.5%	–	–	0.7%/2.1%	1.9%/6.2%	3.8	0.327	6–7

We start with the symmetric network shown in Fig. 6 and assign a capacity of 20 to each link. We define a total of 45 routes with offered loads ranging from 1.0 to 3.0 in such a way that the offered loads at each link in the three peer groups are the same and all transit routes use only one link in the peer groups that they pass through. Each accepted connection generates a revenue of 1.0. Under these conditions, the calculated implied costs c and c' are the same, and, as a result, the revenue sensitivities are also the same. For each link in the peer groups, $c_j = 0.015$. For the links connecting the peer groups, $c_j = 0.129$. Compared to our alternative approximation, the differences are quite small: $\|(c-\tilde{c})/c\|_\infty = 0.7\%$, and $\|(s-\tilde{s})/s\|_\infty = 0.04\%$.

Next, we take the symmetric case and increase the load on the links in peer group 1 to near capacity by increasing the offered loads for local routes in peer group 1 to three and a half times their previous values. This causes the implied cost calculations for c and c' to differ slightly, and, due to the heavy loads in peer group 1, the implied costs $\tilde{c}$ are not as accurate: $\|(c-\tilde{c})/c\|_\infty = 9.0\%$, and $\|(s-\tilde{s})/s\|_\infty = 97.5\%$. (Despite the latter result, we note that $\mathbb{E}[(s-\tilde{s})/s] = \sum_{r\in\mathscr{R}}(v_r(s_r-\tilde{s}_r)/s_r)/\sum_{r\in\mathscr{R}} v_r$ is only 15.0%.) [10] To demonstrate the change in revenue sensitivities from the symmetric case, consider the two alternative routes consisting of the following sets of links:

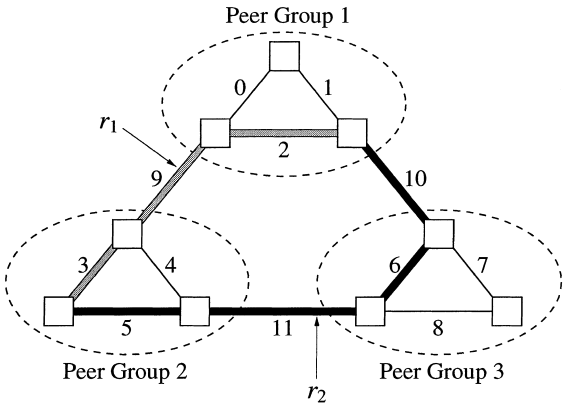


Fig. 6. Symmetric network with a single level of aggregation.

$r_1 = \{2,9,3\}$ and $r_2 = \{10,6,11,5\}$. In the symmetric case, the revenue sensitivities for r_1 and r_2 are 0.823 and 0.684, respectively. In the present overloaded case, the revenue sensitivities change to approximately 0.416 and 0.772, respectively. [11] The longer route is now favored because it avoids passing through the overloaded peer group. We note that, using our first hierarchical approximation, the revenue sensitivity may vary along a particular route depending on which link is making the calculation (due to the $s_{r,j}$ term). To be exact, all links of a route in a given peer group will compute the same sensitivity, but links of the route

[10] Similarly, we define $\mathbb{E}[(c-\tilde{c})/c] = \sum_{j\in\mathscr{J}}(\rho_j(c_j-\tilde{c}_j)/c_j)/\sum_{j\in\mathscr{J}}\rho_j$.

[11] The revenue sensitivity values presented in this section are computed using the surplus values s. Using s' or $\tilde{s}$ results in slightly different values but the same relative ordering.

in a different peer group may compute a different value. For our current example, the revenue sensitivities vary only slightly along routes, on the order of 0.004 in the worst case.

As another example of an overload scenario, we start with the symmetric case and increase the loads on transit routes between peer groups 1 and 2 by one and a half times, causing link 9 to be near capacity. For this case, the differences between the first two approximations are greater than in the previous overload scenario, but the surplus values $\tilde{s}$ fare much better. This is due to the fact that the overloaded node consists of only a single link, mitigating the errors due to local averaging of transit route costs. The revenue sensitivities for r_1 and r_2 are approximately 0.335 and 0.686, respectively, which would cause the routing algorithm to send more traffic around the overload as desired. Compared to the previous case, there is greater variation in the revenue sensitivities along each route using s', on the order of 0.013 in the worst case.

For a fourth experiment with a more varied topology, we use the network shown in Fig. 2. We define a total of 122 routes with offered loads ranging from 0.1 to 2.0. Two routes are defined between each pair of switches except for the members of peer group 2 which have only one local route between each pair. As before, each accepted connection generates a revenue of 1.0. The link capacities are varied between peer groups: links in peer groups 1, 2, and 3 have capacities 25, 40, and 30, respectively, and the connecting links have a capacity of 35 each. Despite the loss of symmetry, the implied cost calculations are surprisingly close.

Table 1 summarizes the main results of the four experiments. $L_{\max}$ is the maximum route blocking probability; the high values for the middle two experiments are for a local route in peer group 1 and a transit route from peer group 1 to 2, respectively. The iterations column denotes the range of iterations needed for convergence of the three implied cost computations. Note that the light load condition $\|\delta\|_M < 1$ holds in every case.

Two comments on the above experiments are in order. First, using our first hierarchical approximation scheme, one can unfortunately construct cases where the revenue sensitivities vary enough along a route to cause an ordering between alternative routes from the source's point of view that is different from that obtained in a flat network. This would cause the adaptive routing algorithm to temporarily shift offered loads in the wrong direction until the sensitivities became farther apart. As a result, the routing algorithm would adapt more slowly, but it is unclear whether this is a common or troubling situation. Second, the bound in Theorem 3 appears to be rather weak. It was too high by an order of magnitude in the two overload cases. In the fourth experiment, however, it was less than twice the actual value.

We also performed experiments on the larger network shown in Fig. 7 with a variable number of defined groups. The group memberships in terms of the links in each group are listed in Table 2. We define a total of 247 routes with offered loads ranging from 0.2 to 3.0. As before, each accepted connection generates a revenue of 1.0. The link capacities vary from 20 to 30, and no attempt was made to equalize the offered loads on the links.

Table 3 summarizes the main results of these six experiments. In terms of relative implied cost and revenue sensitivity errors, the six groups case performed the best, and the six alternate groups and nine groups performed the worst. For these experiments (with fixed routes and offered loads), the error results seem to be correlated to the number of transit routes per group with a lower average number of transit routes tending to produce better results. We also compute the number

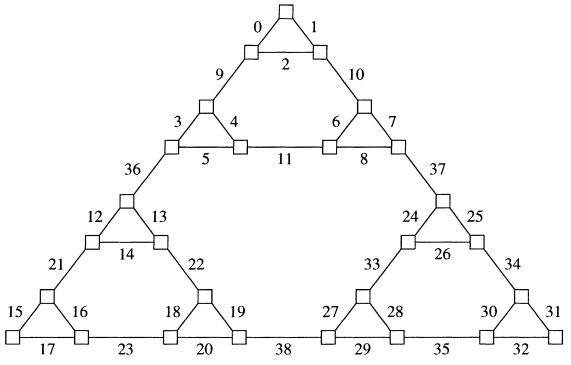


Fig. 7. A larger symmetric network.

Table 2
Group memberships for the experiments on the larger network

3 Groups	{0–11, 36} {12–23, 38} {24–35, 37}
6 Groups	{0–11} {12–23} {24–35} {36} {37} {38}
6 Alternate groups	{0–2, 9–10} {3–8, 11, 36} {12–14, 18–20, 22, 38} {15–17, 21, 23} {24–29, 33, 37} {30–32, 34–35}
9 Groups	{0–2, 9} {3–5, 11, 36} {6–8, 10} {12–14, 21} {15–17, 23} {18–20, 22, 38} {24–26, 33, 37} {27–29, 35} {30–32, 34}
12 Groups	{0–2, 9} {3–5, 11} {6–8, 10} {12–14, 21} {15–17, 23} {18–20, 22} {24–26, 33} {27–29, 35} {30–32, 34} {36} {37} {38}
21 Groups	{0–2} {3–5} {6–8} {9} {10} {11} {12–14} {15–17} {18–20} {21} {22} {23} {24–26} {27–29} {30–32} {33} {34} {35} {36} {37} {38}

Table 3
Computational results for the larger network

	Revenue sensitivity error: $\mathbb{E}[\cdot] / \|\cdot\|_\infty$		Implied cost error: $\mathbb{E}[\cdot] / \|\cdot\|_\infty$		Messages per iteration	Average transit routes per group	Average local routes per group
	$(s - s')/s$	$(s - \tilde{s})/s$	$(c - c')/c$	$(c - \tilde{c})/c$			
3 Groups	3.7%/63.9%	2.8%/120.1%	0.7%/2.9%	1.6%/5.9%	303	14.7	75.0
6 Groups	0.3%/12.2%	0.7%/16.2%	0.05%/0.3%	1.7%/3.9%	312	12.2	36.5
6 Alternate groups	6.8%/159.1%	7.1%/163.1%	1.9%/4.5%	4.9%/8.3%	234	49.0	18.0
9 Groups	10.1%/136.8%	6.9%/98.4%	4.0%/9.6%	4.3%/9.1%	249	43.9	9.7
12 Groups	7.7%/48.6%	4.1%/46.7%	4.5%/8.9%	4.2%/7.7%	294	35.1	7.0
21 Groups	2.7%/13.5%	2.5%/13.5%	1.0%/2.9%	2.2%/8.2%	447	31.0	3.1

of messages per iteration under the assumption that the groups of three switches in a triangle are connected locally using a broadcast medium, i.e., only one message is required to reach the three link controllers in the triangle. For a flat network, 807 messages per iteration are required, so each group structure tested provides a significant reduction. The most savings occurs with the six alternate groups and the nine groups which, as noted above, provide the worst performance in terms of revenue sensitivity error.

6. On-line measurements

We now return to the subject of on-line measurements, as briefly mentioned in Section 3. Instead of using the Erlang fixed point approximation, we show how estimates of the carried loads and blocking probabilities can be used to implement a hierarchical adaptive routing scheme. Our discussion follows that of Kelly [22], with additional optimizations to take advantage of the hierarchical framework.

We say that two routes have the same *hierarchical path* from the point of view of link j if they use the same set of links in peer group $n(j)$ and follow the same sequence of peer groups outside of $n(j)$. Let $\mathscr{H}_n$ be the set of hierarchical paths from the point of view of node n, and let H_{jh} be the amount of bandwidth used explicitly by hierarchical path $h \in \mathscr{H}_n$ on link j. (H_{jh} is 0 for all links j outside of n.) If we make the assumption that $w_{r_1} = w_{r_2}$ for two routes r_1 and r_2 with the same hierarchical structure from the point of view of link $j \in r_1, r_2$, then $s_{r_1;j} = s_{r_2;j}$. Recalling that $\rho_j(1 - B_j) = \sum_{r \in \mathscr{R}} A_{jr}\lambda_r$ and $\delta_j = \eta_j \rho_j$, we can rewrite (11) as

$$c_j = \delta_j \sum_{h \in \mathscr{H}_{n(j)}} H_{jh} \frac{\text{flow carried on path } h}{\text{flow carried through link } j} \times (s_{h;j} + c_j), \quad j \in \mathscr{J}. \tag{22}$$

Suppose we have on-line measures $\hat{\Lambda}_h(t)$ and $\hat{\Theta}_j(t)$ of the carried flows on path h and link j, respectively, over the interval $[t, t+1)$. Smoothed, moving-average estimates $\hat{\lambda}_h(t)$ and $\hat{\theta}_j(t)$ of the mean carried flows can be computed using the iterations

$$\hat{\lambda}_h(t+1) = (1-\gamma)\hat{\lambda}_h(t) + \gamma\hat{\Lambda}_h(t),$$
$$\hat{\theta}_j(t+1) = (1-\gamma)\hat{\theta}_j(t) + \gamma\hat{\Theta}_j(t),$$

where $\gamma \in (0,1)$. If we consider link j to be in isolation with Poisson traffic offered at rate ρ_j, we can estimate ρ_j (and thus δ_j) by solving the equation $\hat{\theta}_j = \rho_j[1 - E(\rho_j, C_j)]$ to obtain $\hat{\rho}_j$. Then we would have $\hat{\delta}_j = \hat{\rho}_j[E(\hat{\rho}_j, C_j - 1) - E(\hat{\rho}_j, C_j)]$.

Now suppose that the implied costs $\hat{c}$ and the associated surplus values $\hat{s}$ have been computed using these estimates and successive substitution. Suppose also that the blocking probability L_h has been estimated for each hierarchical path, possibly using a moving-average estimate similar to the above. The revenue sensitivity $(1-\hat{L}_h)\hat{s}_{h;j}$ tells us the net expected revenue that a call on path h will generate from the perspective of link j. Traffic from a source to a given destination peer group should be split among the possible hierarchical paths based on these revenue sensitivities. A greater share of the traffic should be offered to a path that has a higher value of $(1-\hat{L}_h)\hat{s}_{h;j}$ than the others. Also, if $(1-\hat{L}_h)\hat{s}_{h;j}$ is negative for a particular path, that path should not be used since a net loss in revenue would occur by accepting connections on that path. Any adjustments of the splitting should be done gradually to prevent sudden congestion. Note that we have assumed that routes not satisfying the QoS constraints of a particular connection will be eliminated prior to choosing a path based on the revenue sensitivities.

7. Multiservice extensions

To accommodate different types of services, our model can be extended to a multirate loss network. Now we allow $A_{jr} \in \mathbb{Z}^+$. Several additional problems arise in this context. First and foremost, the Erlang B formula no longer suffices to compute the blocking probability at a link for each type of call. Let $\pi_j(n)$ denote the steady-state probability of n circuits being in use at link j. Then the blocking probability for route r at link j is $B_{jr} = \sum_{n=C_j-A_{jr}+1}^{C_j} \pi_j(n)$. We can compute π_j using a recursive formula of complexity $O(C_jK_j)$ where K_j denotes the number of traffic classes (distinct values of $A_{jr} > 0$) arriving at link j [44]. This result was derived independently by Kaufman and Roberts. To reduce complexity, many asymptotic approximations have been proposed in the literature as the offered load and link capacity are scaled in proportion [18,29,36,42,45,47]. We have found the refined uniform asymptotic approximation (RUAA) developed by Mitra et al. [36] to be particularly accurate.

The Erlang fixed point approximation can be extended in a straightforward manner to the multiservice case using an appropriate blocking function at each link. Note that, in this case, the fixed point is no longer guaranteed to be unique [44]. [12] Based on this approximation, implied cost equations can be derived [11,35], where we now have a different implied cost at each link for each type of service. The straightforward extension to our hierarchical setting is to further compute an average implied cost for each type of service passing through each peer group. Computing a single average implied cost for each peer group is attractive but would probably result in an unacceptable loss in accuracy.

Define $\mathscr{S}$ to be the set of services offered by the network and partition $\mathscr{R}$ into sets $\mathscr{R}^s, s \in \mathscr{S}$. Let $s(r)$ denote the service type associated with route r. [13] Also, let $\rho_{jr} = \lambda_r/(1-B_{jr})$, and define $\eta_{jrq} = B_{jr}(\vec{\rho}_j, \vec{A}_j, C_j - A_{jq}) - B_{jr}(\vec{\rho}_j, \vec{A}_j, C_j)$, which is the expected increase in blocking probability at link j for route r given that A_{jq} circuits are removed from link j. The multiservice implied costs satisfy the following system of equations:

[12] Using a certain single-link blocking function, convergence to a unique fixed point was recently proved in the light load regime only [47].

[13] Note that when multiple service types are carried between two points, we assign various routes that may follow the same path.

$$c_{jq} = \sum_{r:j\in r} \eta_{jrq}\rho_{jr}(s_{r:j} + c_{jr}), \quad j \in \mathscr{J}, \ q \in \mathscr{R}, \tag{23}$$

where

$$s_{r:j} = w_r - \sum_{k\in r} P_{kj}c_{kr} - \sum_{n\neq n(j)} T_{nr}\bar{c}_{ns(r)} \tag{24}$$

and

$$\bar{c}_{ns} = \frac{\sum_{r\in\mathscr{R}^s} T_{nr}\lambda_r(\sum_{j\in r} E_{jn}c_{jr})}{\sum_{r\in\mathscr{R}^s} T_{nr}\lambda_r}. \tag{25}$$

Note that $c_{jr} = c_{jq}$ if $A_{jr} = A_{jq}$. In a large capacity network, we can further reduce (23) to a system of only $3J$ equations by employing the RUAA [36]. If we redefine our norm on $\mathbb{R}^{JR}$ (R is the total number of routes) as

$$\|x\|_M = \max_{j,r:j\in r}\left\{ \sum_{k\neq j:k\in r} P_{kj}|x_{kr}| + \sum_{n\neq n(j)} T_{nr}\overline{|x|}_{ns(r)} \right\}, \tag{26}$$

let $\delta = (\delta_{11}, \delta_{12}, \ldots, \delta_{1R}, \delta_{21}, \ldots, \delta_{JR})$ where $\delta_{jq} = \sum_{r:j\in r} \eta_{jrq}\rho_{jr}$, and define

$$\Delta = \max_{n,r}\left\{ T_{nr}\sum_{m\neq n} T_{mr}|c_r^m - \bar{c}_{ms(r)}| \right\}$$

where $c_r^m = \sum_{j\in r} E_{jm}c_{jr}$, then Theorems 1–3 can be easily shown to hold for the multiservice case.

8. Conclusion

This paper is based on the premise that the use of hierarchical source routing is a key to both reducing complexity and providing acceptable QoS in a large-scale network. Although aggregating network elements into subnetworks is an old idea, we have taken a unique approach to representing the "available" capacity of a subnetwork by formulating an implicit representation based on the average implied cost to go through or into the subnetwork. This average implied cost reflects the congestion in the subnetwork and captures the interdependencies among traffic streams, a feature sorely lacking in explicit representations of available capacity.

We proved that both a synchronous and asynchronous distributed computation of the approximate implied costs will converge to a unique solution under a light load condition. Furthermore, we presented a more aggressive averaging mechanism that also performs local averaging among routes transiting through or into a local subnetwork. We proved that with sufficient damping, a synchronous distributed computation of these new approximate implied costs will converge to a unique solution under any traffic conditions. Our experimental results showed that these approximations are reasonably accurate.

Based on this representation for available subnetwork capacity, we proposed a hierarchical source routing algorithm that adaptively selects high-level routes so as to maximize network revenue. Prior to path selection, routes not likely to meet prespecified QoS constraints, such as end-to-end delay, are eliminated from consideration. Our scheme can incorporate on-line measurements, and it can be extended to a multiservice environment. The low-level routing within subnetworks was deliberately not specified, as we feel that some form of dynamic routing would be beneficial in coping with traffic fluctuations at that level.

Possible topics for future research directly related to our routing algorithm include the following: extensions to more than two levels of hierarchy, the optimal subnetwork size and switch arrangement to achieve the best tradeoff between accuracy and reduced overheads [28], the robustness of the implied costs and routing to link failures, investigation of the need to reserve capacity for local traffic using trunk reservation, and the role of our algorithm in a layered approach to IP over ATM routing [10].

Acknowledgements

This work was supported by a National Science Foundation Graduate Research Fellowship, a Du Pont Graduate Fellowship in Electrical Engineering, a National Science Foundation Career Grant NCR–9624230, and by Southwestern Bell Co.

Appendix A. Proof of Theorem 1

Choose $x, x' \in \mathbb{R}^J$. Then, $\forall j \in \mathscr{J}$,

$$f_j(x) - f_j(x') = -\eta_j(1-B_j)^{-1}\sum_{r\in\mathscr{R}} A_{jr}\lambda_r \times \left(\sum_{k\neq j} A_{kr}P_{kj}(x_k - x'_k) + \sum_{n\neq n(j)} T_{nr}(\bar{x}_n - \bar{x}'_n)\right).$$

Therefore

$$\begin{aligned}|f_j(x) - f_j(x')| &\leqslant \eta_j(1-B_j)^{-1}\sum_{r\in\mathscr{R}} A_{jr}\lambda_r \times \left(\sum_{k\neq j} A_{kr}P_{kj}|x_k - x'_k| + \sum_{n\neq n(j)} T_{nr}|\bar{x}_n - \bar{x}'_n|\right)\\ &\leqslant \eta_j(1-B_j)^{-1}\sum_{r\in\mathscr{R}} A_{jr}\lambda_r\|x - x'\|_M\\ &= \eta_j\rho_j\|x - x'\|_M.\end{aligned}$$

Taking the norm on both sides, we have

$$\|f(x) - f(x')\|_M \leqslant \|\delta\|_M\|x - x'\|_M.$$

So $f(\cdot)$ is a contraction mapping if $\|\delta\|_M < 1$. Using the definition of a contraction mapping and the properties of norms, one can easily show that the sequence $f^i(x), i = 1, 2, \ldots$, converges to c', the unique solution of (11), for any $x \in \mathbb{R}^J$. □

Appendix B. Proof of Theorem 2

Rewrite (11) in matrix form as $f(x) = Gx + b$. The goal is to show that G corresponds to a weighted maximum norm contraction. For, in that case, we can satisfy the conditions of the Asynchronous Convergence Theorem in [5] (see Sections 6.2 and 6.3, pp. 431–435), which guarantees asynchronous convergence to the unique fixed point c'. In the following, we use δ as the weight vector for the weighted maximum norm; in order to do so, we require the condition $\delta > 0$. (We are guaranteed that $\delta \geqslant 0$, but in all practical cases $\delta > 0$ as we have assumed.)

Choose $x, x' \in \mathbb{R}^J$. Then, $\forall j \in \mathscr{J}$,

$$|f_j(x) - f_j(x')| \leqslant \eta_j(1-B_j)^{-1}\sum_{r\in\mathscr{R}} A_{jr}\lambda_r \times \left(\sum_{k\neq j} A_{kr}P_{kj}|x_k - x'_k| + \sum_{n\neq n(j)} T_{nr}|\bar{x}_n - \bar{x}'_n|\right).$$

Therefore

$$\begin{aligned}\left|\frac{f_j(x) - f_j(x')}{\delta_j}\right| &\leqslant \frac{\eta_j(1-B_j)^{-1}}{\delta_j} \times \sum_{r\in\mathscr{R}} A_{jr}\lambda_r\left(\sum_{k\neq j} A_{kr}P_{kj}\delta_k\left|\frac{x_k - x'_k}{\delta_k}\right| + \sum_{n\neq n(j)} T_{nr}\frac{\sum_{l\in\mathscr{J}} E_{ln}(\sum_{q\in\mathscr{R}} T_{nq}A_{lq}\lambda_q)\delta_l\left|\frac{x_l - x'_l}{\delta_l}\right|}{\sum_{q\in\mathscr{R}} T_{nq}\lambda_q}\right)\\ &\leqslant \frac{\eta_j(1-B_j)^{-1}}{\delta_j}\sum_{r\in\mathscr{R}} A_{jr}\lambda_r\left(\sum_{k\neq j} A_{kr}P_{kj}\delta_k + \sum_{n\neq n(j)} T_{nr}\frac{\sum_{l\in\mathscr{J}} E_{ln}(\sum_{q\in\mathscr{R}} T_{nq}A_{lq}\lambda_q)\delta_l}{\sum_{q\in\mathscr{R}} T_{nq}\lambda_q}\right)\|x - x'\|_\infty^\delta\end{aligned}$$

since the weighted maximum norm $\|x\|_\infty^\delta = \max_{j\in\mathscr{J}} |x_j/\delta_j|$. Taking the norm on both sides, we have

$$\|f(x) - f(x')\|_\infty^\delta \leqslant \|G\|_\infty^\delta\|x - x'\|_\infty^\delta,$$

where the induced matrix norm $\|G\|_\infty^\delta = \max_{j\in\mathscr{J}}\{\frac{1}{\delta_j}\sum_{k\in\mathscr{J}}|g_{jk}|\delta_k\}$ [5]. So G corresponds to a weighted maximum norm contraction if $\|G\|_\infty^\delta < 1$. This follows from $\|\delta\|_M < 1$ because

$$\begin{aligned}\|G\|_\infty^\delta &= \max_{j\in\mathscr{J}}\frac{\eta_j(1-B_j)^{-1}}{\delta_j}\sum_{r\in\mathscr{R}} A_{jr}\lambda_r \times \left(\sum_{k\neq j} A_{kr}P_{kj}\delta_k + \sum_{n\neq n(j)} T_{nr}\frac{\sum_{l\in\mathscr{J}} E_{ln}(\sum_{q\in\mathscr{R}} T_{nq}A_{lq}\lambda_q)\delta_l}{\sum_{q\in\mathscr{R}} T_{nq}\lambda_q}\right)\\ &\leqslant \max_{j\in\mathscr{J}}\frac{\eta_j(1-B_j)^{-1}}{\delta_j}\sum_{r\in\mathscr{R}} A_{jr}\lambda_r\|\delta\|_M = \|\delta\|_M\end{aligned}$$

since $\rho_j = (1-B_j)^{-1}\sum_{r\in\mathscr{R}} A_{jr}\lambda_r$ and $\delta_j = \eta_j\rho_j$. □

Appendix C. Proof of Theorem 3

We have, $\forall j \in \mathscr{J}$,

$$c'_j - c_j = \eta_j(1-B_j)^{-1}\sum_{r\in\mathscr{R}} A_{jr}\lambda_r \times \left(\sum_{k\neq j} A_{kr}P_{kj}(c_k - c'_k) + \sum_{n\neq n(j)} T_{nr}(c^n_r - \bar{c}'_n)\right).$$

Hence

$$|c'_j - c_j| \leqslant \eta_j (1 - B_j)^{-1} \sum_{r \in \mathscr{R}} A_{jr} \lambda_r \times \left(\sum_{k \neq j} A_{kr} P_{kj} |c_k - c'_k| + \sum_{n \neq n(j)} T_{nr} |c^n_r - \bar{c}_n + \bar{c}_n - \bar{c}'_n| \right) \leqslant \eta_j \rho_j (\|c' - c\|_M + \Delta). \quad (27)$$

Taking the M-norm on both sides and rearranging, we have

$$\|c' - c\|_M \leqslant \frac{\Delta \|\delta\|_M}{1 - \|\delta\|_M}. \quad (28)$$

We also have, $\forall j, r$ such that $j \in r$,

$$s_r - s'_{r;j} = \sum_{k \in \mathscr{J}} A_{kr} P_{kj} (c'_k - c_k) + \sum_{n \neq n(j)} T_{nr} (\bar{c}'_n - c^n_r).$$

Hence

$$\begin{aligned} |s_r - s'_{r;j}| &\leqslant \sum_{k \in \mathscr{J}} A_{kr} P_{kj} |c'_k - c_k| + \sum_{n \neq n(j)} T_{nr} |\bar{c}'_n - \bar{c}_n + \bar{c}_n - c^n_r| \\ &\leqslant |c'_j - c_j| + \|c' - c\|_M + \Delta \ \ (\text{since } A_{jr} = 1) \\ &\leqslant \eta_j \rho_j (\|c' - c\|_M + \Delta) + \|c' - c\|_M + \Delta \ \ (\text{using (27)}) \\ &= (\delta_j + 1)(\|c' - c\|_M + \Delta) \\ &\leqslant (\delta_j + 1) \frac{\Delta}{1 - \|\delta\|_M} \ \ (\text{using (28)}). \end{aligned}$$

Taking the maximum norm on both sides, the result follows. □

References

[1] A. Alles, ATM internetworking, Cisco Systems, Inc. white paper (http://www.cisco.com/warp/public/614/12.html), May 1995.

[2] ATM Forum, Private Network–Network Interface Specification Version 1.0, ftp://ftp.atmforum.com/pub/approved-specs/af-pnni-0055.000.pdf, March 1996.

[3] N.G. Bean, F.P. Kelly, P.G. Taylor, Braess' paradox in a loss network, Journal of Applied Probability 34 (1) (1997) 155–159.

[4] D. Bertsekas, R. Gallager, Data Networks, 2nd ed., Prentice-Hall, Englewood Cliffs, NJ, 1992.

[5] D. Bertsekas, J. Tsitsiklis, Parallel and Distributed Computation: Numerical Methods, Prentice-Hall, Englewood Cliffs, NJ, 1989.

[6] R. Braden, L. Zhang, S. Berson, S. Herzog, S. Jamin, Resource ReSerVation Protocol (RSVP): Version 1 Functional Specification, RFC 2205, September 1997.

[7] J.E. Cohen, C. Jeffries, Congestion resulting from increased capacity in single-server queueing networks, IEEE/ACM Transactions on Networking 5 (2) (1997) 305–310.

[8] D.E. Comer, Internetworking with TCP/IP: Principles, Protocols, and Architecture, 2nd ed., vol. I, Prentice-Hall, Englewood Cliffs, NJ, 1991.

[9] G. de Veciana, G. Kesidis, J. Walrand, Resource management in wide-area ATM networks using effective bandwidths, IEEE Journal on Selected Areas in Communications 13 (6) (1995) 1081–1090.

[10] P. Dumortier, Toward a new IP over ATM routing paradigm, IEEE Communications 36 (1) (1998) 82–86.

[11] A. Faragó, S. Blaabjerg, L. Ast, G. Gordos, T. Henk, A new degree of freedom in ATM network dimensioning: Optimizing the logical configuration, IEEE Journal on Selected Areas in Communications 13 (7) (1995) 1199–1206.

[12] V. Fayet, D.A. Khotimsky, A. Przygienda, Hop-by-hop routing with node-dependent topology information, in: Proceedings of the IEEE Infocom, 1999.

[13] M.R. Garey, D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, W.H. Freeman, New York, 1979.

[14] R.J. Gibbens, P.A. Whiting, An investigation of the accuracy of implied cost methods for circuit-switched network optimization, in: Fifth UK Teletraffic Symposium, 1988.

[15] A. Girard, Routing and Dimensioning in Circuit-Switched Networks, Addison-Wesley, Reading, MA, 1990.

[16] R. Guérin, A. Orda, QoS routing in networks with inaccurate information: theory and algorithms, IEEE/ACM Transactions on Networking 7 (3) (1999) 350–364.

[17] F. Hao, E.W. Zegura, On scalable QoS routing: performance evaluation of topology aggregation, in: Proceedings of the IEEE Infocom, 2000.

[18] J.Y. Hui, Switching and Traffic Theory for Integrated Broadband Networks, Kluwer Academic, Boston, 1990.

[19] C. Huitema, Routing in the Internet, Prentice-Hall, Englewood Cliffs, NJ, 1995.

[20] F. Kamoun, L. Kleinrock, Stochastic performance evaluation of hierarchical routing for large networks, Computer Networks 3 (5) (1979) 337–353.

[21] F.P. Kelly, Blocking probabilities in large circuit-switched networks, Advances in Applied Probability 18 (2) (1986) 473–505.

[22] F.P. Kelly, Routing in circuit-switched networks: optimization, shadow prices, and decentralization, Advances in Applied Probability 20 (1) (1988) 112–144.

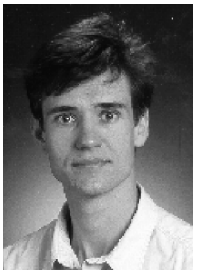
[23] F.P. Kelly, Loss networks, Annals of Applied Probability 1 (3) (1991) 319–378.

[24] F.P. Kelly, Network routing, Philosophical Transactions of the Royal Society of London, Series A 337 (1647) (1991) 343–367.

[25] F.P. Kelly, Notes on effective bandwidths, in: F.P. Kelly, S. Zachary, I.B. Ziedins (Eds.), Stochastic Networks: Theory and Applications, Oxford University Press, New York, 1996, pp. 141–168.

[26] P.B. Key, Implied cost methodology and software tools for a fully connected network with DAR and trunk reservation, British Telecom Technology Journal 6 (1988) 52–65.
[27] L. Kleinrock, F. Kamoun, Hierarchical routing for large networks, Computer Networks 1 (3) (1977) 155–174.
[28] R. Krishnan, R. Ramanathan, M. Steenstrup, Optimization algorithms for large self-structuring networks, in: Proceedings of the IEEE Infocom, 1999.
[29] J.-F.P. Labourdette, G.W. Hart, Blocking probabilities in multitraffic loss systems: insensitivity, asymptotic behavior, and approximations, IEEE Transactions on Communications 40 (8) (1992) 1355–1366.
[30] C. Labovitz, G.R. Malan, F. Jahanian, Internet routing instability, IEEE/ACM Transactions on Networking 6 (5) (1998) 515–528.
[31] W.C. Lee, Spanning tree method for link state aggregation in large communication networks, in: Proceedings of the IEEE Infocom, vol. 1, 1995, pp. 297–302.
[32] W.C. Lee, Topology aggregation for hierarchical routing in ATM networks, Computer Communication Review 25 (2) (1995) 82–92.
[33] W.C. Lee, M.G. Hluchyj, P.A. Humblet, Routing subject to quality of service constraints in integrated communication networks, IEEE Network 9 (4) (1995) 46–55.
[34] D.H. Lorenz, A. Orda, QoS routing in networks with uncertain parameters, IEEE/ACM Transactions on Networking 6 (6) (1998) 768–778.
[35] D. Mitra, J.A. Morrison, K.G. Ramakrishnan, ATM network design and optimization: a multirate loss network framework, IEEE/ACM Transactions on Networking 4 (4) (1996) 531–543.
[36] D. Mitra, J.A. Morrison, K.G. Ramakrishnan, Optimization and design of network routing using refined asymptotic approximations, Performance Evaluation 36/37 (1999) 267–288.
[37] D. Mitra, J.A. Morrison, K.G. Ramakrishnan, Virtual private networks: joint resource allocation and routing design, in: Proceedings of the IEEE Infocom, 1999.
[38] M. Montgomery, Managing complexity in large-scale networks via flow and network aggregation, Ph.D. thesis, The University of Texas at Austin, August 1998.
[39] M. Montgomery, G. de Veciana, Hierarchical source routing through clouds, in: Proceedings of the IEEE Infocom, vol. 2, 1998, pp. 685–692.
[40] A. Orda, Routing with end-to-end QoS guarantees in broadband networks, IEEE/ACM Transactions on Networking 7 (3) (1999) 365–374.
[41] N.S.V. Rao, S.G. Batsell, QoS routing via multiple paths using bandwidth reservation, in: Proceedings of the IEEE Infocom, vol. 1, 1998, pp. 11–18.
[42] J. Roberts, U. Mocci, J. Virtamo (Eds.), Broadband Network Teletraffic: Performance Evaluation and Design of Broadband Multiservice Networks, Final Report of Action COST 242, Springer, Berlin, 1996.
[43] R. Rom, PNNI routing performance: an open issue, in: Washington University Workshop on Integration of IP and ATM, (http://www.arl.wustl.edu/arl/workshops/atmip/proceedings.html), November 1996.
[44] K.W. Ross, Multiservice Loss Models for Broadband Telecommunication Networks, Springer, London, 1995.
[45] A. Simonian, J.W. Roberts, F. Théberge, R. Mazumdar, Asymptotic estimates for blocking probabilities in a large multi-rate loss network, in: Proceedings of the 33rd Annual Allerton Conference on Communication, Control, and Computing, 1995, pp. 726–735.
[46] M.E. Steenstrup (Ed.), Routing in Communication Networks, Prentice-Hall, Englewood Cliffs, NJ, 1995.
[47] F. Théberge, R.R. Mazumdar, New reduced load heuristic for computing blocking in large multirate loss networks, IEE Proceedings: Communications 143 (4) (1996) 206–211.
[48] Z. Wang, J. Crowcroft, Quality-of-service routing for supporting multimedia applications, IEEE Journal on Selected Areas in Communications 14 (7) (1996) 1228–1234.
[49] W.-L. Yang, Estimation and abstraction of available capacity in large-scale networks, Master's thesis, The University of Texas at Austin, 1997.
[50] L. Zhang, S. Deering, D. Estrin, S. Shenker, D. Zappala, RSVP: a new resource ReSerVation Protocol, IEEE Network 7 (5) (1993) 8–18.

Michael Montgomery received the B.Sc. degree in Computer Engineering and the M.Sc. degree in Electrical Engineering from Virginia Tech, Blacksburg, VA, in 1993 and 1994, respectively. He received a Ph.D. in Electrical and Computer Engineering from the University of Texas at Austin in 1998. He is currently a research staff member with the Center for Information Infrastructure Technology at the Oak Ridge Y-12 Plant, where he works on projects in the areas of mobile ad hoc networks, QoS routing, distributed computing environments, and telemedicine.



Gustavo de Veciana received his B.Sc., M.Sc., and Ph.D. in electrical engineering from the University of California at Berkeley in 1987, 1990, and 1993 respectively. In 1993, he joined the Department of Electrical and Computer Engineering at the University of Texas at Austin where he is currently an Associate Professor. His research focuses on issues in the design and control of telecommunication networks. Dr. de Veciana is an editor for the IEEE/ACM Transactions on Networking. He is the recipient of a General Motors Foundation Centennial Fellowship in Electrical Engineering and a 1996 National Science Foundation CAREER Award.