Hierarchical Source Routing Through Clouds

MICHAEL MONTGOMERY* AND GUSTAVO DE VECIANA†

Department of Electrical and Computer Engineering
The University of Texas at Austin
Austin, Texas 78712–1084

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 cloud (subnetwork), we compute the average implied cost to go through or into the cloud. Such implied costs reflect the congestion in the cloud 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. To assess accuracy, we derive a bound on the difference between our implied costs and those calculated for a flat network. In addition, we show how on-line measurements can be incorporated into the routing algorithm, and we present some representative computational results which demonstrate the ability of our scheme to appropriately route high level flows while significantly reducing complexity.

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, allowing one to properly provision for QoS. Similarly, QoS extensions to the Internet, such as RSVP [7, 18], make such networks akin to connection-oriented technologies. Indeed, the idea 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 [10]). Similar comments apply to an IP over ATM switching environment, where IP flows are mapped to ATM virtual circuits. In light of the above trends 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. After first introducing hierarchical source routing, we explain the basics of our routing algorithm and give an example of the complexity reduction that it can achieve.

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 nicely into the ATM PNNI (Private Network-Network Interface) framework [17], but it can also be thought of as a candidate for replacing the Border Gateway Protocol (BGP) [7] in the Internet that would split flows in "IP/RSVP" routing. Central to our algorithm is the *implied cost* [9] of a connection along a given path which measures the expected increase in future blocking that would occur from accepting this connection. Using implied costs takes into account the possibility of "knock-on" effects (due to blocking and subsequent alternate routing) [9] and results in a *system optimal* routing algorithm.

To make good decisions and provide acceptable QoS, it is desirable to have a global view of the network at the source when making routing decisions for new connections. Thus, source routing, where the source specifies the entire path for the connection, is an attractive routing method. It has the additional advantage that, in contrast to hop-by-hop routing, there is no need to run a standardized routing algorithm to avoid loops and policy issues such as provider selection are easily accommodated. Propagating information for each link throughout the network quickly becomes unmanageable as the size of the network increases, so a hierarchical structure is needed, such as that proposed in the ATM PNNI specification [17]. 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 for 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 also makes addressing feasible in a large-scale network, as demonstrated by the network addressing of IP, and it permits the use of different routing schemes at different levels of the hierarchy. Prior work in the area of routing in networks with inaccurate information can be found in [5].

By combining a hierarchical network with (loose[1]) source routing, we have a form of routing referred to as *hierarchical*

*M. Montgomery is supported by a National Science Foundation Graduate Research Fellowship and a Du Pont Graduate Fellowship in Electrical Engineering. E-mail: mcm@mail.utexas.edu

†G. de Veciana is supported by a National Science Foundation Career Grant NCR-9624230 and by Southwestern Bell Co. Tel: (512) 471–1573 Fax: (512) 471–5532 E-mail: gustavo@ece.utexas.edu

[1] 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 node of that peer group.

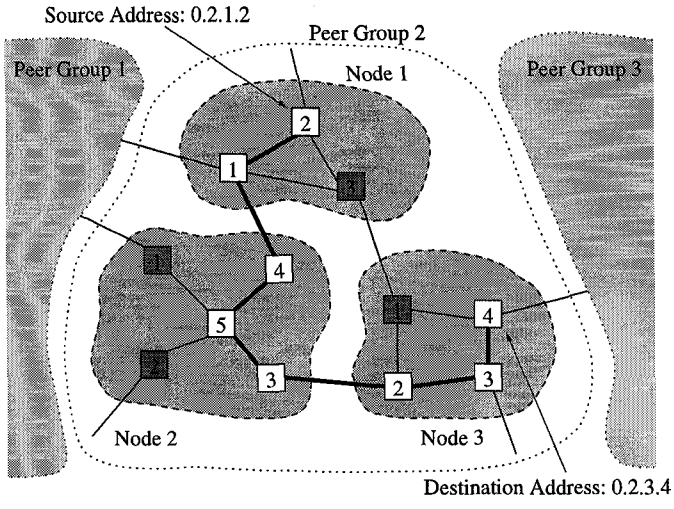

Figure 1: Illustration of hierarchical addressing and source routing.

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.[2] 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 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., $0.2.1.2 \rightarrow 0.2.1.1 \rightarrow 0.2.2 \rightarrow 0.2.3$. 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. (Our model will ignore crankback.)

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 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 [12]. 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 [12]. Most proposed solutions lie somewhere between these two extremes. 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 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 accepting a call would have on the congestion level both within the peer group and in other parts of the network due to interdependencies among traffic streams. 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.

Such implied costs reflect the congestion in peer groups as well as the interdependencies among traffic streams in the network, and they may be useful to network operators for the purpose of assessing current congestion levels. 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. 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 propagation delay,[3] 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 because of the availability of accurate local routing information.

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.

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

[2]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.

[3]Queueing delays are assumed to be small and are ignored.

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, 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 information update in an algorithm such as PNNI is a real problem, as it can easily overload the network elements [15].

The rest of this paper is organized as follows. Section 2 explains our model and some notation. The theoretical basis of our adaptive routing scheme and its relation to Kelly's work is given in Section 3. Section 4 presents some computational results. In Section 5, we discuss on-line measurements of some necessary parameters, and Section 6 briefly outlines extensions to a multi-service environment.

2 Model and notation

Our model is that of a loss network serving a single type of traffic, 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.[4] The capacity of each link $j \in \mathcal{J}$ is C_j circuits, 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 a node n is defined as a collection of links that form a peer group or that connect two peer groups. 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 the same node). A route is considered to be a collection of links from $\mathcal{J}$; route $r \in \mathcal{R}$ uses A_{jr} circuits on link $j \in \mathcal{J}$, where $A_{jr} \in \{0,1\}$. 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 ν_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 as, for example, is 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 will 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.

[4] One realistic example of a single-service environment is a single-class embedded network. Alternatively, our model is roughly equivalent to a network with very high bandwidth links where the real resource constraint is that of labels (e.g., virtual path or virtual circuit identifiers) for connections on links. The unit bandwidth requirement per call can be considered to be an *effective bandwidth* [2, 11].

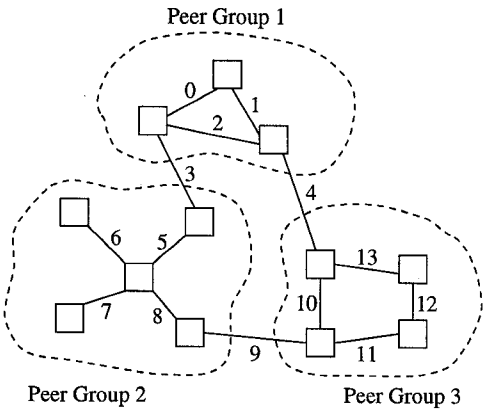

Figure 2: Example network with a single level of aggregation.

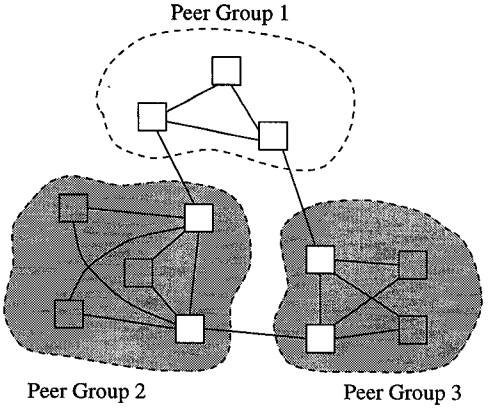

Figure 3: Logical view of the network from the perspective of peer group 1.

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 approximated, the customary method being the Erlang fixed point [4, 10].

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

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

where

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

and the function E is the Erlang B formula

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

The vector B is called the Erlang fixed point; its existence follows from the Brouwer fixed point theorem and uniqueness was proved in [8]. Using B, an approximation for the blocking probability on route r is

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

The idea behind the approximation is as follows. Each Poisson stream of rate ν_r that passes through link j is thinned by a factor $1 - B_k$ at each link $k \in r - \{j\}$ before being offered to j. If these thinnings were independent both from link to link and over all routes (this is not really true), then the traffic offered to link j would be Poisson with rate ρ_j, as given in (2), B_j, from (1), would be the blocking probability at link j, and L_r, from (4), would be the exact loss probability on route r.

Due to the on-line nature of our algorithm, we feel that instead of using the Erlang fixed point to approximate the blocking probabilities, it will be more accurate and efficient to measure the relevant quantities. Specifically, L_r, λ_r (the throughput achieved on route r), and $\sum_{r \in \mathcal{R}} A_{jr} \lambda_r$ (the total throughput through link j) will be calculated based on moving-average estimates. This will in turn allow us to compute the associated implied costs and surplus values and hence the approximate revenue sensitivities.

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

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

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,[5] the sensitivity of the rate of revenue with respect to the offered loads has been derived by Kelly [9] and is given by

$$\frac{\partial}{\partial \nu_r} W(\nu; C) = (1 - L_r) s_r \tag{6}$$

where

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

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

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

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 $\lambda_r = \nu_r (1 - L_r)$.

[5] 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.

Remark. 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 $j \in r$. The implied costs c quantify the knock-on effects due to accepting a call. 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. We note that, in general, $W(\nu; C)$ is not concave. However, Kelly has shown that it is asymptotically linear as ν and C are increased in proportion [9]. Furthermore, even though a hill-climbing algorithm could potentially reach a non-optimal local maximum, 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 node n (or, equivalently, over the links in n visited by such routes) where, in the following, $c_r^n = \sum_{j \in \mathcal{J}} A_{jr} E_{jn} c_j$:

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

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* $j \in r$:

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

The link implied costs are now calculated as

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

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 Kelly's implied costs.

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 $f = (f_1, f_2, \ldots, f_J)$,

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

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.

Define a norm on $\mathbb{R}^J$ by

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

where

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

For any positive vector α, we define the weighted maximum norm on $\mathbb{R}^J$ by $||x||_\infty^\alpha = \max_j |\frac{x_j}{\alpha_j}|$, 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.

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

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

Therefore

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

Taking the norm on both sides, we have

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

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 $x \in \mathbb{R}^J$. □

Remark. The product $\eta_j\rho_j$ increases to 1 as ρ_j, the offered load at link j, increases [9]. So $||\delta||_M < 1$ can be referred to as a light load condition. If the network has long routes and/or heavily loaded links, it 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 example, 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 Thm. 1 assumes iterates are computed synchronously. In a large-scale network, synchronous computation is infeasible, so we will show that our light load condition is sufficient for convergence of an asynchronous computation in the following sense [1]:

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 $t \ge t_2$, 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. 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 [1] (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 (note that $\delta \ge 0$, but in all practical cases $\delta > 0$ as we have assumed).

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

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

Therefore

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

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

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

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

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

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

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* $c_r^m = \sum_{j\in\mathcal{J}}A_{jr}E_{jm}c_j$*. Then we have*

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

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

Proof. We have, $\forall j \in \mathcal{J}$,

$$\begin{aligned}c'_j - c_j &= \eta_j(1-B_j)^{-1}\sum_{r\in\mathcal{R}}A_{jr}\lambda_r(\sum_{k\neq j}A_{kr}P_{kj}(c_k - c'_k) \\ &\quad + \sum_{n\neq n(j)}T_{nr}(c_r^n - \bar{c}'_n)).\end{aligned}$$

Hence

$$\begin{aligned}|c'_j - c_j| &\leq \eta_j(1-B_j)^{-1}\sum_{r\in\mathcal{R}}A_{jr}\lambda_r(\sum_{k\neq j}A_{kr}P_{kj}|c_k - c'_k| \\ &\quad + \sum_{n\neq n(j)}T_{nr}|c_r^n - \bar{c}_n + \bar{c}_n - \bar{c}'_n|) \\ &\leq \eta_j\rho_j(||c' - c||_M + \Delta).\end{aligned} \tag{15}$$

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

$$||c' - c||_M \leq \frac{\Delta||\delta||_M}{1 - ||\delta||_M}. \tag{16}$$

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

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

Hence

$$\begin{aligned}|s_r - s'_{r;j}| &\leq \sum_{k\in\mathcal{J}}A_{kr}P_{kj}|c'_k - c_k| + \sum_{n\neq n(j)}T_{nr}|\bar{c}'_n - \bar{c}_n + \bar{c}_n - c_r^n| \\ &\leq |c'_j - c_j| + ||c' - c||_M + \Delta \qquad \text{since } A_{jr} = 1 \\ &\leq \eta_j\rho_j(||c' - c||_M + \Delta) + ||c' - c||_M + \Delta \qquad \text{using (15)} \\ &= (\delta_j + 1)(||c' - c||_M + \Delta) \\ &\leq (\delta_j + 1)\frac{\Delta}{1 - ||\delta||_M} \qquad \text{using (16).}\end{aligned}$$

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

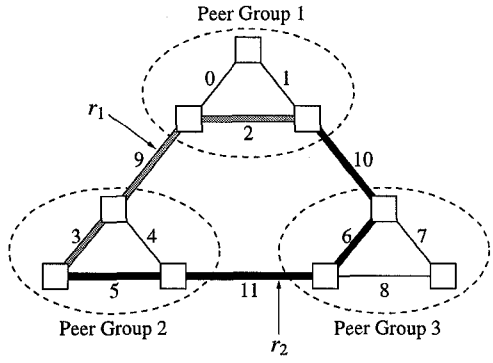

Figure 4: Symmetric network with a single level of aggregation.

Remark. The error between our modified implied costs and Kelly's implied costs will be minimized under light loads ($||\delta||_M \ll 1$) and if the difference between transit route costs in 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.

4 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 calculation. We start with the symmetric network shown in Fig. 4 and assign a capacity of 20 to each link. We have defined 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 are the same using either approximation, i.e., $||s - s'||_\infty = 0$, 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, $c_j = 0.129$. We also note that the maximum blocking probability for a route is 2.1% and $||\delta||_M = 0.297$.

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 to differ slightly, resulting in $||c - c'||_\infty = 0.0004$ and $||s - s'||_\infty = 0.007$. The maximum blocking probability for a route is now 25% (for a local route in peer group 1), and $||\delta||_M = 0.764$. To demonstrate the change in revenue sensitivities from the previous case, consider the two alternative routes consisting of the fol-

lowing sets of links: $r_1 = \{2,9,3\}$ and $r_2 = \{10,6,11,5\}$. In the symmetric case, the revenue sensitives 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. The longer route is now favored because it avoids passing through the overloaded peer group. We note that, using our 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 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. As expected, the results are similar to the previous case with slightly greater differences between the two approximations due to the greater global effect of the overload: $\|c - c'\|_\infty = 0.006$, $\|s - s'\|_\infty = 0.026$, the maximum blocking probability for a route is 16% (for a transit route from peer group 1 to 2), and $\|\delta\|_M = 0.780$. The revenue sensitivities for r_1 and r_2 are approximately 0.335 and 0.686, respectively. There is greater variation in the revenue sensitivities along each route, on the order of 0.013 in the worst case.

For a final 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 with the exception of the members of peer group 2 having 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: $\|c - c'\|_\infty = 0.002$ and $\|s - s'\|_\infty = 0.110$. The maximum blocking probability for a route is 3.8%, and $\|\delta\|_M = 0.327$.

Two comments on the above experiments are in order. First, 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. Secondly, the bound in Thm. 3 appears to be rather weak. It was too high by an order of magnitude in the two overload cases. In the last experiment, however, it was less than twice the actual value.

5 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 [9], 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 $\mathcal{H}_n$ be the set of hierarchical paths from the point of view of peer group n, and let H_{jh} be the amount of bandwidth used explicitly by hierarchical path $h \in \mathcal{H}_n$ 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 $\rho_j(1 - B_j) = \sum_{r \in \mathcal{R}} A_{jr}\lambda_r$ and $\delta_j = \eta_j \rho_j$, we can rewrite (11), for $j \in \mathcal{J}$, as

$$c_j = \delta_j \sum_{h \in \mathcal{H}_{n(j)}} H_{jh} \frac{\text{flow carried on path } h}{\text{flow carried through link } j}(s_{h;j} + c_j). \quad (17)$$

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.

6 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 $O(C_j K_j)$ where K_j denotes the number of traffic classes (distinct values of $A_{jr} > 0$) arriving at link j [16]. 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 [6, 13, 14]. We have found Mitra and Morrison's Uniform Asymptotic Approximation (UAA) [13] 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 [16]. Based on this approximation, implied cost equations can be derived [3, 13], 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 $\mathcal{S}$ to be the set of services offered by the network and partition $\mathcal{R}$ into sets $\mathcal{R}^s, s \in \mathcal{S}$. Let $s(r)$ denote the service type associated with route r.[6] 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:

$$c_{jq} = \sum_{r:j\in r} \eta_{jrq}\rho_{jr}(s_{r;j} + c_{jr}), \quad j \in \mathcal{J},\ q \in \mathcal{R}, \tag{18}$$

where

$$s_{r;j} = w_r - \sum_{k\in r} P_{kj} c_{kr} - \sum_{n\neq n(j)} T_{nr}\bar{c}_{ns(r)} \tag{19}$$

and

$$\bar{c}_{ns} = \frac{\sum_{r\in\mathcal{R}^s} T_{nr}\lambda_r(\sum_{j\in r} E_{jn}c_{jr})}{\sum_{r\in\mathcal{R}^s} T_{nr}\lambda_r}. \tag{20}$$

Note that $c_{jr} = c_{jq}$ if $A_{jr} = A_{jq}$. In a large capacity network, we can further reduce (18) to a system of only J equations by employing the UAA [13]. If we redefine our norm on $\mathbb{R}^{JR}$ (R is the total number of routes) as

$$||x||_M = \max_{j,r:j\in r}\{\sum_{k\neq j:k\in r} P_{kj}|x_{kr}| + \sum_{n\neq n(j)} T_{nr}\overline{|x|}_{ns(r)}\}, \tag{21}$$

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 $\Delta = \max_{n,r}\{T_{nr}\sum_{m\neq n} T_{mr}|c_r^m - \bar{c}_{ms(r)}|\}$ where $c_r^m = \sum_{j\in r} E_{jm}c_{jr}$, then Thms. 1, 2, and 3 can be easily shown to hold for the multiservice case.

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

References

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

[2] G. de Veciana, G. Kesidis, and J. Walrand, "Resource management in wide-area ATM networks using effective bandwidths," *IEEE Journal on Selected Areas of Communications*, vol. 13, no. 6, pp. 1081–1090, Aug. 1995.

[3] A. Faragó, S. Blaabjerg, L. Ast, G. Gordos, and T. Henk, "A new degree of freedom in ATM network dimensioning: Optimizing the logical configuration," *IEEE Journal on Selected Areas of Communications*, vol. 13, no. 7, pp. 1199–1206, Sept. 1995.

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

[5] R. Guérin and A. Orda, "QoS-based routing in networks with inaccurate information: Theory and algorithms," in *Proc. IEEE Infocom*, 1997.

[6] J. Y. Hui, *Switching and Traffic Theory for Integrated Broadband Networks*, Boston: Kluwer Academic Publishers, 1990.

[7] C. Huitema, *Routing in the Internet*, Englewood Cliffs, NJ: Prentice Hall, 1995.

[8] F. P. Kelly, "Blocking probabilities in large circuit-switched networks," *Advances in Applied Probability*, vol. 18, no. 2, pp. 473–505, June 1986.

[9] F. P. Kelly, "Routing in circuit-switched networks: Optimization, shadow prices, and decentralization," *Advances in Applied Probability*, vol. 20, no. 1, pp. 112–144, Mar. 1988.

[10] F. P. Kelly, "Loss networks," *The Annals of Applied Probability*, vol. 1, pp. 319–378, 1991.

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

[12] W. C. Lee, "Spanning tree method for link state aggregation in large communication networks," in *Proc. IEEE Infocom*, vol. 1, pp. 297–302, 1995.

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

[14] J. Roberts, U. Mocci, and J. Virtamo, eds., *Broadband Network Teletraffic: Performance Evaluation and Design of Broadband Multiservice Networks; Final Report of Action COST 242*, Berlin: Springer-Verlag, 1996.

[15] R. Rom, "PNNI routing performance: An open issue," in *Washington University Workshop on Integration of IP and ATM*, Nov. 1996.

[16] K. W. Ross, *Multiservice Loss Models for Broadband Telecommunication Networks*, London: Springer-Verlag, 1995.

[17] The ATM Forum, "Private network-network interface specification version 1.0," Mar. 1996.

[18] L. Zhang, S. Deering, D. Estrin, S. Shenker, and D. Zappala, "RSVP: A new resource ReSerVation Protocol," *IEEE Network*, vol. 7, pp. 8–18, Sept. 1993.