Guaranteeing End-to-end Quality of Service in Connection-oriented Packet Networks

A. Weinrib and R. Nagarajan

COINS Technical Report 91-51

Guaranteeing End-to-end Quality of Service in Connection-oriented Packet Networks

Abel Weinrib
Bellcore
445 South Street, MRE 2D-298
Morristown, NJ 07962-1910

Ramesh Nagarajan [1]
Dept. of Electrical and Computer Engineering
Univ. of Massachusetts
Amherst, MA 01003

Abstract

Data communication networks of today offer little guarantees on the quality (grade) of service provided to subscribers. Advances in transmission and switching technology are expected to usher in new applications in future packet communication networks, which will require that these networks provide explicit service guarantees to subscribers. In this paper we investigate resource allocation policies for satisfying *end-to-end* quality of service (QOS) guarantees in a connection-oriented (virtual circuit) network. Our goal is to maximize the network efficiency (i.e., maximize the number of connections that can be carried while meeting the end-to-end guarantee) by allocating differing amounts of resources at the different nodes of the network, thus minimizing the impact of bottleneck nodes where resources are rare. For a number of exactly-solvable models, we find the optimal policy and show that a simple policy (requiring each of the intermediate nodes on the source-destination path to contribute equally to meeting the end-to-end performance) can be significantly sub-optimal. For systems where less knowledge of the connection parameters is available, we propose an adaptive heuristic policy, and through simulation show that it performs well. Finally, we develop a measure of the benefit to be expected from non-uniform resource allocation, thus explaining the different behaviors observed for the various models.

[1]Most of this work was done while Ramesh Nagarajan was employed at Bellcore during the summer of 1989. He was also partially supported under DARPA grant NAG2-578 at U. Mass.

1 Introduction

Current packet communication networks offer users very little, in terms of guaranteed quality of service (QOS), above a "best-effort" delivery of information. If the goal of providing a variety of services on a single integrated public packet network is to become a reality, future networks will need to provide explicit end-to-end service guarantees to subscribers. Due to the availability of high bandwidth fiber and advanced switching technology, future integrated networks will support a variety of novel applications, such as broadcast TV, voice and high speed data transfers. These applications will require that a subscriber be able to specify for a connection an *a priori* quality of service, expressed as constraints on the end-to-end delay of information units, the acceptable levels of information loss, and so on. The network must then be capable of guaranteeing these performance constraints for the duration of the connection.

In this paper, we concentrate on the problem of guaranteeing *end-to-end* quality of service in connection-oriented packet networks. It is clear that to guarantee performance, network resources must be allocated to connections. We address the question of how best to allocate network resources to different connections so as to maximize overall network efficiency. Network efficiency is measured by the total number of connections that can be supported by the network while still providing all the QOS guarantees. When there is an imbalance in the network resulting in "bottleneck" nodes where resources are scarce (and thus valuable), we shall see that it pays to meet the required end-to-end guarantees by requiring less stringent local guarantees from bottleneck nodes while compensating with more stringent local guarantees at other nodes of the network. We find that the optimal allocation of network resources can be significantly more efficient than a simple scheme that allocates resources uniformly throughout the network. By allowing for non-uniform requirements at the nodes, less valuable resources at less congested nodes can be exchanged for the more valuable resources at bottleneck nodes.

In related work, other authors have discussed providing end-to-end performance guarantees; see, e.g., [FV90] and [CY89]. Ferrari and Verma [FV90] consider a scheme for establishing "real-time channels" with guaranteed end-to-end delay. Their scheme requires that the end-to-end delay be assigned to the various nodes, and thus fits the general framework that we develop in Section 2; however, they do not discuss the problem of how to assign the local guarantees across nodes, and choose to use a simple heuristic policy (see their Equations (14) and (15)).

Other related work has concentrated on only a *single* switch or node in the network. For instance, Nain and Ross [NR86] consider a system with multiple classes of traffic. They investigate policies that minimize the average delay for all classes except one class whose average delay is constrained to be below a specified value. Viniotis and Ephremides [VE88] consider a system with a mixture of voice and data traffic. They study policies that minimize a weighted sum of the blocking probability for voice calls and the average delay for data messages. Chipalkatti et al. [C^+89] evaluate scheduling disciplines for a mixture of real-time and non-real-time traffic so as to flexibly trade-off the delay for non-real-time traffic against the loss of real-time traffic (due to expiration of a deadline).

Since this previous work considers only the performance at a single node, it is complementary to our focus on the efficient allocation of network-wide resources to provide end-to-end QOS guarantees to individual connections that pass through many nodes. In this paper, we do not address how to provide service guarantees at a single node. Instead, we assume that such policies are available (i.e., that the function $g(r)$ introduced in the next section is known), and examine policies that translate the end-to-end service guarantee (e.g., end-to-end packet delay) to local requirements at each node (e.g., nodal packet delay) through which the connection passes.

The local guarantees assigned by a policy then imply some allocation of resources at the various nodes. Since there are finite network resources, there is a maximum number of connections that can be admitted under a given policy. The policies are ranked according to their network efficiency as measured by this maximum number of connections that each policy admits. The policies are also compared according to the total amount of resources allocated to each connection.

The simplest policy is one that equally distributes the burden of satisfying the guarantee among the nodes of the connection. We shall see that such a simple policy can be *significantly* sub-optimal as measured by network efficiency (the number of connections admitted), even though this policy minimizes the amount of resources per connection. In maximizing the number of connections, the optimal policy uses more total resources per connection, but less of the valuable bottleneck-node resources. The optimal policy makes use of complete knowledge of the types and mixture of connections, as well as the network state. We also introduce and study the performance of a heuristic adaptive policy for use when knowledge of connection types is unavailable. While the heuristic policy allocates less total resources per connection than the optimal policy, it nevertheless achieves high network efficiency,

In Section 2, we introduce our general model. To demonstrate the value of our proposed non-uniform allocation of network resources to connections, we use some simple instances of the model for which the optimally efficient allocation of network resources can be found exactly. Section 3 examines the case that a connection's average packet delay is guaranteed, and Section 4 considers the case that a connection's packet loss probability is guaranteed.

It would be nice to have an *a priori* predictor for the potential benefit of our proposed non-uniform resource-allocation scheme. In Section 5 we introduce the Relative Gain Ratio (RGR) that provides us with this information. The RGR explains the results of the earlier sections: when the quality of service is average delay, substantial improvements in network efficiency are realized with the optimal policy, while for guaranteed packet loss probability only small improvements are possible.

2 General Model

Figure 1 illustrates our model network, showing a single source-destination pair of nodes between which connections are routed. This simple model captures the essence of the problem, and allows us to exactly solve for various optimal allocation policies. Interfering network side-traffic that arises in more general networks is modeled by appropriately reduced node resources. Thus, a "bottleneck" node in this model is one with few available resources. In this network model, only one route exists between the source and destination nodes; we do not address routing of connections, implicitly assuming that the route is determined *a priori*. (While not included in this paper, route selection clearly provides an important additional degree of freedom for resource allocation in a real network, and must be included in a complete solution of the problem.)

Connection requests arrive at the source node with specified QOS criteria. Each connection is admitted or rejected by the allocation (admission) policy which guarantees the end-to-end QOS by assigning fractions of the allowed QOS among the nodes of the connection. Over time, connection requests arrive and connections terminate dynamically. In this paper, we do not study this dynamic process, but rather study the best that each policy can achieve: for a given policy and network, N is the maximum number of connections that can be admitted. Our goal is to compare the different policies based on N, which measures how efficiently the total network resources are being used to satisfy the connection requests. For convenience, we shall allow non-integer N; in reality the number of connections will be the largest integer less than N.

We now introduce some of the notation used in this paper. Let C be the number of traffic types or classes, and $G^{(c)}$ denote the end-to-end QOS to be guaranteed for that class, $c = 1, \ldots, C$. The source-destination route is taken to be h hops long. Each node i ($i = 1, \ldots, h$) along the connection has a finite amount, R_i, of some resource to be shared among the connections routed through the node. For a node i, let $r_i^{(c)}$ be the resource allocated to an arbitrary connection of type c routed through this node. While in the calculations that follow we consider only one type of resource at a time, more generally there may be a number of different types of resources (e.g., bandwidth, buffer space, etc.), in which case R_i and $r_i^{(c)}$ will be multi-component vectors. For conciseness, the notation introduced here only allows for policies that perform the same allocation of resources to all connections of the same type; we shall have to slightly modify our notation when we introduce our adaptive policy that allocates differing amounts of resources to the same connection type. Also, we will omit the superscripts pertaining to the connection type when we consider a single connection class.

We assume that we have available the function $g_i(r_i)$ which gives the QOS guaranteed by node i to a connection that has been allocated r_i resources at the node. Then, the QOS constraint to be satisfied for all connection types is

$$\Gamma(g_1(r_1^{(c)}),\ g_2(r_2^{(c)}), \ldots,\ g_h(r_h^{(c)})) \leq G^c \quad c = 1, \ldots, C \tag{1}$$

where Γ is a function that captures the aggregation of the individual nodal guarantees. For instance, if the QOS guarantee G is the end-to-end delay and the g_is are the node delays, then Γ simply sums its arguments.

Let $N^{(c)}$ be the number of connections of type c admitted by some policy, so that $N = \sum_{c=1}^{C} N^{(c)}$. The resource constraint to be satisfied at each node i is that the total resources allocated can not exceed the available resources,

$$\sum_{c=1}^{C} N^{(c)} r_i^{(c)} \leq R_i \quad i = 1, \ldots, h \tag{2}$$

The set of equations given by (1) and (2) define our general model. To solve for the optimal allocation policy OPT, we need to find the allocations $r_i^{(c)}$ (resources at node i allocated to a connection of type c) that maximizes N, while satisfying the constraint equations. We contrast the performance of this optimal policy with the equal guarantee policy EG. The EG policy assigns an equal amount of the QOS for a connection to each node, so $g_i(r_i^{(c)}) = g_j(r_j^{(c)}) \quad \forall i, j$. If $g_i()$ is independent of i, then the EG policy allocates equal resources at each node to a connection.

Since we are looking for the maximum number of connections that can be supported subject to the resource and QOS constraints, the constraint equations (1) and (2) are taken to be equalities (rather than inequalities) in further discussion. Clearly, N can be increased for any policy unless the end-to-end QOS is satisfied exactly, and unless all the resources at at least one node are utilized. The optimal policy will use all nodal resources, while other policies may leave some resources idle at non-bottleneck nodes.

Another quantity of interest is the total network resources allocated to each connection type,

$$r^{(c)}_{total} = \sum_{i=1}^{h} r^{(c)}_i \quad c = 1, \ldots, C \tag{3}$$

Somewhat counter-intuitively, minimizing the total resources per connection does *not* in general maximize the network efficiency (as measured by the maximum number of connections admitted N) because simply minimizing the total resources does not account for the bottleneck nodes. In fact, we conjecture that the EG policy, which we shall find performs poorly in some cases, will minimize $r^{(c)}_{total}$. With the reasonable assumptions that the functions $g_i(r)$ are independent of i and are convex in r, allocation of equal resources at each of the nodes will minimize r_{total}, since a reduction by Δr in the resources used at one node will require more than Δr resources at some other nodes to compensate. Appendix A contains a proof of this conjecture for the guaranteed-delay model of the next section.

For the examples of this paper, we assume that packets arrive according to a Poisson process and that the packet lengths are exponentially distributed with the same parameter (for convenience only) for all connection types. These assumptions result in simple forms for the function $g()$, and hence allow us to derive exact solutions.

In the next section, we take the node resource to be the bandwidth (link capacity) and the QOS, G, to be the end-to-end average packet delay. Then we consider the situation where the node resource is buffer space (memory) and the end-to-end QOS is the packet loss probability.

3 Guaranteeing Average Delay

In this section we investigate models where the QOS is the average packet delay for a connection. We solve for the performance of the optimal (OPT) and equal-guarantee (EG) policies, and introduce a new adaptive (AD) policy for these models. In the final subsection we graphically compare and contrast the results for the various policies and models. We study two cases of the model of Section 2: with two hops and two connection classes ($h = 2$, $C = 2$), and with h hops and a single connection class ($C = 1$). For these models the nodal resource is the bandwidth, with link capacities C_i playing the role of R_i at node i, and the connections are characterized by their packet arrival rate $\lambda^{(c)}$ and their allowable average delay $d^{(c)}$.

A straightforward way to guarantee a connection's average delay at a node, independent of the behavior of the other connections, is to assign adequate bandwidth to the connection and impose a processor-sharing type of service discipline such as Fair Queueing [DKS89] or Virtual Clock [Zha89] that guarantees fair-allocation of the bandwidth to connections. A connection's traffic is assumed to be Poisson with rate $\lambda^{(c)}$ and the packet lengths are assumed to be exponentially distributed (with parameter 1.0 for convenience). Thus, if r units of bandwidth are allocated to a connection at some node, then the nodal guarantee will be given by the $M/M/1$ formula,

$$g(r) = \frac{1}{r - \lambda} \tag{4}$$

Note that the form of the function is the same for all nodes (independent of i).

3.1 Two hops, two connection types

With two hops ($h = 2$) and two connection types ($C = 2$), we wish to determine the optimal OPT resource allocation scheme and also compare its performance to that of the simple equal-guarantee EG policy.

Since there are two connection types, we need to address the issue of traffic mix. With no constraint on the traffic mix, the optimal policy will maximize the number of connections by simply admitting only the connection type with a lower value of $\lambda + \frac{1}{d}$ (refer to Appendix B). Thus, the problem reduces to that with only one connection type, and is uninteresting. Instead, we impose a fairness constraint on the traffic mix, requiring that

$$\frac{N^{(1)}}{N^{(1)} + N^{(2)}} = p \quad 0 \leq p \leq 1 \tag{5}$$

for a given value of the traffic-mix parameter p.

3.1.1 Equal-guaranty policy

Here, each node is required to contribute $d^{(c)}/h$ of the total allowable end-to-end delay $d^{(c)}$. The nodal guarantees to be satisfied are,

$$g(r_1^{(c)}) = g(r_2^{(c)}) = \frac{d^{(c)}}{2} \quad c = 1, 2 \tag{6}$$

Inserting $g()$ from equation (4) and solving for the $r_i^{(c)}$, we find

$$\begin{aligned} r_1^{(1)} = r_2^{(1)} &= \lambda^{(1)} + \frac{2}{d^{(1)}} \\ r_1^{(2)} = r_2^{(2)} &= \lambda^{(2)} + \frac{2}{d^{(2)}} \end{aligned} \tag{7}$$

The capacity constraints (Equation (2)) are

$$\begin{aligned} N^{(1)}r_1^{(1)} + N^{(2)}r_1^{(2)} &\leq C_1 \\ N^{(1)}r_2^{(1)} + N^{(2)}r_2^{(2)} &\leq C_2 \end{aligned} \tag{8}$$

The smaller capacity node will be the bottleneck, so

$$N = \frac{\min(C_1,\ C_2)}{pr_1^{(1)} + (1-p)r_1^{(2)}} \tag{9}$$

where we have carried out the replacement $N^{(1)} = pN$ and $N^{(2)} = (1-p)N$. The total bandwidth allocated per connection is given by,

$$r_{total}^{(c)} = 2(\lambda^{(c)} + \frac{2}{d^{(c)}}) \tag{10}$$

3.1.2 Optimal policy

The delay constraints for the optimal policy are,

$$\begin{aligned} g(r_1^{(1)}) + g(r_2^{(1)}) &= d^{(1)} \\ g(r_1^{(2)}) + g(r_2^{(2)}) &= d^{(2)} \end{aligned} \tag{11}$$

and the capacity constraints are,

$$\begin{aligned} N^{(1)}r_1^{(1)} + N^{(2)}r_1^{(2)} &= C_1 \\ N^{(1)}r_2^{(1)} + N^{(2)}r_2^{(2)} &= C_2 \end{aligned} \tag{12}$$

Substituting $N^{(1)} = pN$ and $N^{(2)} = (1-p)N$ in the above equations we have

$$\frac{C_1}{pr_1^{(1)} + (1-p)r_1^{(2)}} = \frac{C_2}{pr_2^{(1)} + (1-p)r_2^{(2)}} \tag{13}$$

We have four variables $r_1^{(1)}$,$r_2^{(1)}$,$r_1^{(2)}$ and $r_2^{(2)}$. The two delay constraints and the equation above constitute three equations in these four variables. The fourth equation to solve for the variables is obtained from the condition for maximizing the total number of connections N. The detailed analysis is presented in Appendix B.

Results for the case of a two hops and a single connection type ($h = 2$, $C = 1$) are immediately obtained by setting $p = 0$ in the results of Appendix B. It can be seen from the analysis that the OPT policy allocates the nodal resources for the connection in the ratio of the link capacities,

$$\frac{r_1}{r_2} = \frac{C_1}{C_2} \tag{14}$$

3.2 Multiple hops, one connection type

Here we generalize the analysis of the previous section for a single connection type to an arbitrary number of hops.

3.2.1 Equal-guaranty policy

The EG policy divides the end-to-end delay d equally among the h nodes of the connection, so

$$g(r_1) = g(r_2) = \ldots = g(r_h) = \frac{d}{h} \tag{15}$$

Hence we have

$$r_1 = r_2 = \ldots = r_h = \lambda + \frac{h}{d} \tag{16}$$

The maximum number of connections admitted by the EG policy is then determined by the minimum-capacity bottleneck node, and

$$N = \min_{i=1,\ldots,h}\left(\frac{C_i}{\lambda + \frac{h}{d}}\right) \tag{17}$$

The total resources allocated by the EG policy to each connection is given by,

$$r_{total} = h\left(\lambda + \frac{h}{d}\right) \tag{18}$$

3.2.2 Optimal policy

The capacity constraints give us h equations,

$$N r_i = C_i \quad i = 1, \ldots, h \tag{19}$$

and the delay constraint gives us one more,

$$\sum_{i=1}^{h} \frac{1}{r_i - \lambda} = d \tag{20}$$

These $h+1$ equations completely determine the $h+1$ unknowns (the r_i and N). Substituting $r_i = \frac{c_i}{c_1} r_1$ from Equation (19) in Equation (20), we have

$$\sum_{i=1}^{h} \frac{1}{\frac{C_i}{C_1} r_1 - \lambda} = d \tag{21}$$

and finding r_1 reduces to finding the roots of a polynomial of order h; the valid solution for r_1 is $r_1 > \lambda + \frac{1}{d}$. The maximum number of connections as well as the total resources allocated per connection is then easy to evaluate.

3.3 Adaptive policy

The resource allocation policies considered so far have been "off-line" policies depending on *a priori* information on traffic classes, mixes, and routes. We now propose a heuristic that may be employed "on-line" for resource allocation when such *a priori* information is unavailable. We assume that the currently available nodal resources are known, and evaluate the heuristic allocation algorithm for each connection request that arrives. The algorithm is adaptive in that it uses the current network state to make its decisions, and thus can adapt to different situations (such as changing traffic mixes, connection types, etc.).

Since no knowledge regarding the connection types and their routing is available, it would be wise to minimize the total resources allocated to each connection so as to prevent the creation of artificial (policy induced) bottlenecks. On the other hand, there are likely to be actual bottlenecks in the network due to the scarcity of resources at some nodes; it would be wise to consume as little of these nodes' resources as possible, even at the expense of a larger consumption at some other node of the connection. Our heuristic incorporates these contrasting principles, giving the amount of resources to be allocated to the current connection request based on the QOS requirement of the request and the state of the nodes.

Define A_i to be the capacity already allocated at each node i, and let r_i be the capacity to be allocated to the current connection request. Consider the following function,

$$S = \sum_{i=1}^{h} \frac{r_i}{C_i - A_i} \tag{22}$$

S is the sum of the resources allocated at each node, weighted by $(C_i - A_i)^{-1}$ which is a measure of the value of the resources at node i to future connection requests. By minimizing the above function (constrained by the quality of service) we will incorporate both the principles outlined earlier.

In Appendix C we derive the resource allocation scheme that minimizes the above function S for the delay model being considered. For an h hop connection the resources allocated at a node i is

$$r_i = \lambda + \frac{1}{d}(\sum_{j=1}^{h} \sqrt{\frac{C_i - A_i}{C_j - A_j}}) \tag{23}$$

The actual capacity allocated depends on both the QOS required by the connection and the residual capacity at each of the nodes, which in turn depends on the previous connections' allocations. Hence, unlike the OPT and EQ policies, connections of the same type (with the same QOS requirements) may be allocated different amounts of bandwidth.

Further, it can be shown that:

$$\frac{r_i - \lambda}{r_j - \lambda} = \sqrt{\frac{C_i - A_i}{C_j - A_j}} \quad \forall i, j \tag{24}$$

The above result is similar in spirit (i.e., resources allocated in ratio of "available" capacities) to the resource allocation ratio of the "off-line" OPT policy.

3.4 Results

Figures 2, 3, and 4 display results for the guaranteed-average-delay model of this section. We plot the number of connections admitted and the total resources allocated per connection for the optimal (OPT), equal-guarantee (EG), and adaptive (AD) policies.

Figure 2 is for the two-hop, two-connection-type ($h = 2$, $C = 2$) model with $\lambda^{(1)} = 1$, $d^{(1)} = 0.1$, $\lambda^{(2)} = 2$, $d^{(2)} = 0.5$, and equal mix of the two classes of traffic, $p = 0.5$. The capacity of one link is held fixed at $C_2 = 1000$, and the other is varied from 0 to 2500. For $C_1 < C_2$, link #1 is the bottleneck, while link #2 is the bottleneck when $C_1 > C_2$; when $C_1 = C_2$ there are no bottlenecks and the EG scheme performs optimally. In Figure 2(a), which shows the connections admitted for the various schemes, we see that the OPT and AD schemes provide a substantial improvement over the simple EG scheme when there is a bottleneck in the network. Note that the curves are smooth for the OPT and EG policies because we allow non-integer solutions of the constraint equations; the AD curve is discontinuous because N is found by simulation of the policy, so only integer values are obtained.

Figure 2(b) displays the total resources per connection. The OPT policy allocates the largest total resources per connection, while the EG scheme allocates the least; the AD scheme lies between these two policies. In Appendix A we show that the EG policy actually minimizes the total resource allocated per connection, even when it does not result in good network efficiency. These results illustrates the fact that network efficiency (maximizing the number of connections admitted) comes from trading off less valuable resources at the non-bottleneck node for the more valuable resources at the bottleneck node, even though more total resources are used for a connection.

Figure 3 shows similar results for the delay model with one connection type and $h = 3$ hops; the three-hop model requires solution of a cubic equation, which can be solved exactly. The connection parameters are $\lambda = 1$ and $d = 0.1$, while the first link capacity C_1 is varied with fixed $C_2 = C_3 = 1000$.

Figure 4 is for the two-hop, two-connection-type model with two very different classes of traffic: $\lambda^{(1)} = 2.5$, $d^{(1)} = 4$, $\lambda^{(2)} = 1/7$, $d^{(2)} = 0.7$. The link capacities are $C_1 = 100$ and $C_2 = 1000$, and we vary the traffic-mix parameter p from 0 to 1. We have chosen both classes of traffic to have the same value of $\lambda + \frac{2}{d} = 3$, so the EG policy treats them the same, and the results are independent of the traffic mix. However, the OPT policy shows significant p-dependence: when most of the connections are of class 2 (small p) the OPT policy admits almost twice as many connections as the EG policy, while when most of the connections are of class 1 (large p), the OPT policy offers little improvement over the simple EG policy. The difference between the two classes of traffic is that $\lambda^{(1)}d^{(1)} = 10$ while $\lambda^{(2)}d^{(2)} = 0.1$. In Section 5 we shall show that the value of λd determines the potential performance improvement to be gained from the OPT policy; only for small λd is there much potential for improvement over the EG policy.

4 Guaranteeing Packet Loss

In this section we study the model of Section 2, taking the QOS guarantee to be the packet loss probability, and taking the nodal resource allocated to connections to be buffer space (memory). The analysis is presented for our network model with two hops, $h = 2$, and a single connection type, $C = 1$.

Define b to be the acceptable end-to-end packet-loss probability, and let B_1 and B_2 the total available buffer space at each node (measured in numbers of packets). Loss probabilities of interest are likely to be quite small (e.g., $< 10^{-3}$). Defining the loss probabilities at the nodes to be b_1 and b_2, the end-to-end loss can be approximated as the sum of the nodal loss probabilities,

$$\begin{aligned} b &= 1 - (1 - b_1)(1 - b_2) \\ &\approx b_1 + b_2 \end{aligned} \tag{25}$$

where we neglect the product term of order b^2 on the right hand side of the above expression due to the small values of b.

To derive a tractable model, we make a number of fairly strong simplifying assumptions; the model is introduced primarily to contrast the results with the results for the guaranteed-delay models studied in the previous section. In a complete treatment of the packet-loss model, two types of resources should be considered, buffer space and bandwidth, thus allowing for tradeoffs between the two. Here, we assume that the bandwidth allocated for a connection C_i is an external parameter, fixed in advance, and is the same for all connections and at all nodes. Thus, we introduce as a system parameter the traffic intensity for a connection $\rho \equiv \lambda / C_i$.

Because the loss probability b is small, the intensity of the traffic at the downstream node will be essentially unchanged. However, we shall also assume that the traffic remains Poisson. This assumption will be good if there is some mechanism that re-randomizes the traffic between the first and second node; if not, large bursts which lead to packet loss at the first node will have been thinned out (by the packet loss at that node) before arriving at the second. We would expect this assumption to result in overestimation of the packet loss at the second node, and is thus a conservative approximation.

With the Poisson assumption, the nodal packet loss probability is given by the M/M/1/K formula [Kle75],

$$b_i = g(r_i) \equiv \frac{\rho^{r_i}(1-\rho)}{1-\rho^{r_i+1}} \quad i = 1,2 \tag{26}$$

The QOS constraint equations are

$$g(r_1) + g(r_2) = b \tag{27}$$

and the buffer constraint equations are

$$Nr_1 \leq B_1 \tag{28}$$

$$Nr_2 \leq B_2 \tag{29}$$

where r_i is the buffer space allocated at node i to a connection.

As in the previous section, we examine policies that satisfy the QOS with the above constraints. The EG policy requires each node to equally drop packets. The OPT policy partitions the end-to-end loss probability so as to maximize the number of connections that can be admitted while satisfying the QOS.

4.1 Equal-guarantee policy

Under this policy each node is assigned half of the end-to-end loss probability,

$$b_1 = b_2 = \frac{b}{2} \tag{30}$$

The buffer space to be allocated at each node is derived by inverting the function $g(r)$,

$$g^{-1}(x) = \frac{\log x - \log(1-\rho+x\rho)}{\log \rho} \tag{31}$$

The maximum number of connections that can be admitted under this policy is then,

$$N = \frac{\min(B_1,\ B_2)}{g^{-1}(b/2)} \tag{32}$$

4.2 Optimal policy

For the OPT policy, the end-to-end QOS constraint is

$$\frac{\rho^{r_1}(1-\rho)}{1-\rho^{r_1+1}} + \frac{\rho^{r_2}(1-\rho)}{1-\rho^{r_2+1}} = b \tag{33}$$

and the buffer constraints are

$$Nr_1 = B_1 \tag{34}$$

$$Nr_2 = B_2 \tag{35}$$

which reduces to

$$\frac{r_1}{r_2} = \frac{B_1}{B_2} \tag{36}$$

Substituting equation (36) in equation (33) gives an equation in a single variable. While there seems to be no closed form solution to this equation, it is easily solved numerically.

4.3 Results

Figure 5 displays results for the guaranteed-loss model of this section, showing the number of connections admitted for the optimal (OPT) and equal-guarantee (EG) policies. We take a model with buffer space $B_2 = 1000$ at node #2, and vary the buffer space available at node #1 between 0 and 2500. The load parameter is $\rho = 0.5$. We consider two values for the guaranteed end-to-end packet-loss probability b: Figure 5(a) shows results for relatively large loss probability ($b = 0.1$), and Figure 5(b) for small loss probability ($b = 10^{-4}$). We see that for large loss probability, the OPT policy offers some improvement in network efficiency compared to the EG policy, but for small loss probability little improvement over the EG policy is possible. Unfortunately, we would expect that most applications will require small loss probability, where little is to be gained from non-uniform allocation of resources; this is to be contrasted with the results for the delay model of the previous section, where significants improvements in efficiency were realized in an interesting regime. In the next section, we turn to the problem of predicting the improvement to be expected from using a more efficient policy compared with the simple EG policy.

5 QOS Criteria and Optimal Resource Allocation

In this section, we explore the effect of the particular QOS criteria on the choice of a policy for effective resource allocation. We shall find that the choice of policy depends upon the shape of the function that governs the relationship between the QOS criteria and the resource being allocated. We illustrate this for the cases of the average packet delay QOS and the packet loss probability QOS.

The EG policy divides the end-to-end QOS equally among the nodes of the connection, regardless of the total available resources at the nodes. An intelligent policy partitions the QOS so as to assign to the bottleneck node (with fewer resources) as large a portion of the end-to-end QOS as possible, thus requiring fewer resources at that node. The key issue is thus the reduction in the resources allocated at the bottleneck node that results from a larger (less stringent) QOS constraint. If this reduction is large, then more sophisticated policies will have potential benefit. On the other hand, if the reduction is small, the EG policy may perform close to optimally.

Let G_i be the nodal fraction of the end-to-end QOS assigned to node i, so

$$G_i = g(r_i) \tag{37}$$

We are interested in measuring the change in resources (r_i) required to meet a particular QOS assignment to node i (G_i) as the assignment is varied. The following function, which we shall call the Relative Gain Ratio (RGR), provides us with the required information:

$$\begin{aligned} RGR &= -\frac{dr_i/r_i}{dG_i/G_i} \\ &= -\frac{g(r_i)}{r_i}\frac{1}{\frac{dg(r_i)}{dr_i}} \end{aligned} \tag{38}$$

where the minus sign reflects the fact that fewer resources are required as the assignment of the constraint (G_i) is increased. If the value of RGR is large, then there is significant potential benefit from a policy that optimally assigns the guarantees required of each node (compared to, for e.g., the simple EG policy). If RGR is small, then there is little potential benefit and any reasonable policy will perform close to optimally. We now examine the RGR for the models of Sections 3 and 4, thus explaining the differing improvements in network efficiency from the optimal policies for the two systems.

For the average delay we have,

$$g(r_i) = \frac{1}{r_i - \lambda} \tag{39}$$

so that

$$RGR = \frac{1}{1 + \lambda G_i} \tag{40}$$

The maximum value of the RGR is unity and occurs when $\lambda G_i = 0$. As seen in Figure 6(a), for small λG_i the RGR is close to one, thus explaining the significant difference in the observed performance of the EG and OPT policies of Section 3 for small values of $\lambda^{(c)} d^{(c)}$.

We can carry out a similar analysis for the packet loss probability function. The RGR in this case is

$$RGR = \frac{1 - \rho}{(1 - \rho + G_i \rho) \log(\frac{1 - \rho + G_i \rho}{G_i})} \tag{41}$$

Figure 6(b) shows the value of the RGR in this case for different values of G_i and ρ. We see that the value of the RGR is large only for large packet-loss probability and high traffic intensities, which is not a particularly interesting regime for most applications. In the more interesting regime (considered in Section 4) of small packet loss probability, the RGR value is low and hence the equal guarantee policy does almost as well as the optimal policy.

Thus, we see that a simple evaluation of the RGR provides insight into the policy choice for a given QOS criteria. For systems with small RGR there is little benefit to be gained from using a policy other than EG; however, when the RGR is large, a better policy has significant potential benefit.

6 Conclusions

In this paper, we have considered the issue of resource allocation to provide a guaranteed quality of service in an efficient manner. In particular, we have investigated policies that provide a guaranteed quality of service while maximizing the total number of connections carried by the network. After defining a general model for such systems, we turned to various simple models for which we find the optimal policy exactly. For these models, we have shown that a straightforward policy of sharing the end-to-end quality of service among the nodes on the source-destination path can perform significantly worse then the optimal policy. These results suggest that our proposal to relieve bottleneck nodes by non-uniformly allocating resources at network nodes should be included as part of the connection-admission policy in future networks.

The optimal policies are based on complete knowledge of the network traffic. A heuristic adaptive policy was proposed for situations where this information is unavailable, and this policy has been shown to perform well. Further, we have developed a measure of the potential improvement from the optimal policy compared with a simple policy such as the equal guarantee policy for different QOS criteria.

The major contributions of this paper lie in introducing the idea of non-uniform allocation of resources to provide end-to-end QOS guarantees, in demonstrating that the technique can provide significant improvements in network efficiency, and in defining the *RGR* measure of the potential improvements to be expected from the technique. Many interesting and important problems remain; for instance, future problems to be studied include:

- The behavior of the system as connections are requested and terminated dynamically.
- The design of admission-control protocols that efficiently implement non-uniform allocation of resources to connections.
- The advantage of reallocating resources dynamically in response to the arrival of connection requests or to changing QOS requirements of existing connections.
- A study of the RGR for other systems, and developing optimal or good heuristic algorithms for those systems where non-uniform allocation appears promising.
- The routing of connections, which has been ignored in this paper, will have significant impact on network efficiency. Both the general problem of alternately routing connec-

tions and the interaction of routing with our non-uniform resource-allocation scheme, deserve further investigation.

References

[C+89] R. Chipalkatti et al. Scheduling policies for real-time and non-real-time traffic in a statistical multiplexer. In *IEEE INFOCOM '89*, pages 774–783, 1989.

[CY89] D. Comer and R. Yavatkar. Flows: Performance guarantees in best effort delivery systems. In *IEEE INFOCOM '89*, pages 100–109, 1989.

[DKS89] A. Demers, S. Keshev, and S. Shenker. Analysis and simulation of a fair queueing algorithm. In *ACM SIGCOMM '89*, 1989.

[FV90] D. Ferrari and D. C. Verma. A scheme for real-time channel establishment in wide-area networks. *IEEE J. on Selected Areas in Communications*, 8:368–379, 1990.

[Kle75] L. Kleinrock. *Queueing Systems*, volume 1. John Wiley & Sons, 1975.

[NR86] P. Nain and K. W. Ross. Optimal multiplexing of heterogenous traffic with hard constraint. In *ACM Joint Conf. on Computer Performance Measurement, Modeling and Evaluation*, pages 100–108, 1986.

[PP88] Protter and Protter. *Fourth Edition Calculus with Analytical Geometry*. Jones and Bartlett, 1988.

[VE88] I. Viniotis and A. Ephremides. Optimal switching of voice and data at a network node. In *IEEE INFOCOM '88*, pages 1504–1507, 1988.

[Zha89] L. Zhang. *A New Architecture for Packet Switching Network Protocols*. PhD thesis, Massachusetts Institute of Technology, 1989.

Appendix A

Here we show that the EG policy minimizes the total resources allocated per connection for the delay model of Section 3.2.

The optimal policy problem can be formulated as a constrained minimization problem which is solved by the method of Lagrange multipliers. The goal is to minimize,

$$r_{total} = \sum_{i=1}^{h} r_i \tag{42}$$

Let d_i be the portion of the end-to-end average delay d allocated to node i. Then the delay constraint is

$$\sum_{i=1}^{h} d_i = d \tag{43}$$

and inverting Equation (4), we have

$$r_i = \lambda + \frac{1}{d_i} \tag{44}$$

Using the method of Lagrange multipliers we convert the problem to one of free minimization,

$$r_{total} = h\lambda + \sum_{i=1}^{h} \frac{1}{d_i} + l(\sum_{i=1}^{h} d_i - d) \tag{45}$$

where l is the Lagrange multiplier. Differentiating with respect to the d_i's and l, we find

$$\frac{-1}{d_i^2} + l = 0 \quad i = 1, \ldots, h \tag{46}$$

$$\sum_{j=1}^{h} d_j - d = 0 \tag{47}$$

These $h + 1$ equations are uniquely satisfied by:

$$d_i = \frac{d}{h} \quad i = 1, \ldots, h \tag{48}$$

which is the equal guarantee policy.

Appendix B

We solve for the optimal resource allocation scheme in the case of the delay model of Section 3 with two connection types and a two hop connection. The solution for the case of one connection type and two hops is a special case of the above.

Let us first write down the delay constraints for the model:

$$\frac{1}{r_1^{(1)} - \lambda^{(1)}} + \frac{1}{r_2^{(1)} - \lambda^{(1)}} = d^{(1)} \tag{49}$$

$$\frac{1}{r_1^{(2)} - \lambda^{(1)}} + \frac{1}{r_2^{(2)} - \lambda^{(2)}} = d^{(2)} \tag{50}$$

$$\tag{51}$$

It is important to note that $r_i^{(c)} > \lambda^{(c)} + \frac{1}{d^{(c)}}$ for any capacity allocation scheme to satisfy the delay constraint. We can express the variables $r_2^{(1)}$ and $r_2^{(2)}$ in terms of the other two variables using the above equations,

$$r_2^{(1)} = f^{(1)}(r_1^{(1)}) = \lambda^{(1)} + \frac{1}{d^{(1)} - \frac{1}{r_1^{(1)} - \lambda^{(1)}}} \tag{52}$$

$$r_2^{(2)} = f^{(2)}(r_1^{(2)}) = \lambda^{(2)} + \frac{1}{d^{(2)} - \frac{1}{r_1^{(2)} - \lambda^{(2)}}} \tag{53}$$

Next let us examine the capacity constraints,

$$N^{(1)} r_1^{(1)} + N^{(2)} r_1^{(2)} = C_1 \tag{54}$$

$$N^{(1)} r_2^{(1)} + N^{(2)} r_2^{(2)} = C_2 \tag{55}$$

Substituting $N^{(1)} = pN$ and $N^{(2)} = (1-p)N$ we have,

$$N = C_1/(pr_1^{(1)} + (1-p)r_1^{(2)}) = C_2/(pr_2^{(1)} + (1-p)r_2^{(2)}) \tag{56}$$

Using the substitutions for $r_2^{(1)}$ and $r_2^{(2)}$ we have,

$$N = C_1/(pr_1^{(1)} + (1-p)r_1^{(2)}) = C_2/(pf^{(1)}(r_1^{(1)}) + (1-p)f^{(2)}(r_1^{(2)})) \tag{57}$$

We need to maximize N. Note that the above relation is defined by the intersection of two surfaces O and P (functions of two variables). The intersection of the two surfaces is a space curve [PP88]. To maximize N we need to determine the critical point of this curve. We outline the strategy below.

Define the surfaces

$$O(r_1^{(1)}, r_1^{(2)}) = C_1/(pr_1^{(1)} + (1-p)r_1^{(2)}) \tag{58}$$

$$P(r_1^{(1)}, r_2^{(2)}) = C_2/(pf^{(1)}(r_1^{(1)}) + (1-p)f^{(2)}(r_1^{(2)})) \tag{59}$$

Then vectors normal to O and P, denoted by n_1 and n_2, respectively, are [PP88]:

$$n_1 = (O_{r_1^{(1)}}, O_{r_1^{(2)}}, -1) \tag{60}$$

$$n_2 = (P_{r_1^{(1)}}, P_{r_1^{(2)}}, -1) \tag{61}$$

where the subscripted functions denote partial derivatives with respect to the subscripted quantities, i.e.,

$$O_{r_1^{(1)}} = \frac{\partial O(r_1^{(1)}, r_1^{(2)})}{\partial r_1^{(1)}} \tag{62}$$

The curve that constitutes the intersection of the surfaces must be perpendicular to both the normal vectors since the curve lies in both surfaces. Hence its direction numbers are given by the vector product of n_1 and n_2,

$$n_1 \times n_2 = (\cdots, \cdots, O_{r_1^{(1)}} P_{r_1^{(2)}} - O_{r_1^{(2)}} P_{r_1^{(1)}}) \tag{63}$$

The x and y components are not listed as they are not of interest. The critical points of the curve are at the points that the z-component of the direction vector vanishes,

$$O_{r_1^{(1)}} P_{r_1^{(2)}} - O_{r_1^{(2)}} P_{r_1^{(2)}} = 0 \tag{64}$$

This provides us with the fourth equation required to solve for the unknown resource allocation parameters, and yields, after some manipulation, a simple relation between $r_1^{(1)}$ and $r_1^{(2)}$:

$$r_1^{(1)} = \lambda^{(1)} + \frac{d^{(2)}}{d^{(1)}}(r_1^{(2)} - \lambda^{(2)}) \tag{65}$$

The above equation in conjunction with equation (57) can be used to solve for $r_1^{(2)}$ and then all the other variables. Defining constants,

$$\begin{aligned} k_1 &= C_2(1-p) + C_2 p \frac{d^{(2)}}{d^{(1)}} \\ k_2 &= C_2 p(\lambda^{(1)} - \frac{d^{(2)}}{d^{(1)}}\lambda^{(2)}) \\ k_3 &= C_1 p \frac{d^{(2)}}{d^{(1)}} + C_1 p \lambda^{(1)} d^{(2)} + (1-p)C_1 \lambda^{(2)} d^{(2)} + C_1(1-p) \\ k_4 &= -(C_1 p \lambda^{(1)} \lambda^{(2)} d^{(2)} + C_1 p \lambda^{(1)} + C_1 p \frac{d^{(2)}}{d^{(1)}} \lambda^{(2)} + C_1(1-p)(\lambda^{(2)})^2 d^{(2)} \\ &\quad +2C_1(1-p)\lambda^{(2)}) \end{aligned} \tag{66}$$

$r_1^{(2)}$ satisfies the quadratic equation

$$k_1 d^{(2)}(r_1^{(2)})^2 + (k_2 d^{(2)} - k_1(\lambda^{(2)} d^{(2)} + 1) - k_3) r_1^{(2)} - k_2(\lambda^{(2)} d^{(2)} + 1) - k_4 = 0 \tag{67}$$

The acceptable solution for $r_1^{(2)}$ (out of the two solutions obtained) is the one that is greater than $\lambda^{(2)} + \frac{1}{d^{(2)}}$.

To obtain the solution for the case of a single connection type, set $p = 0$. Then we obtain,

$$\begin{aligned} k_1 &= C_2 \\ k_2 &= 0 \\ k_3 &= C_1(1 + \lambda^{(2)} d^{(2)}) \\ k_4 &= -C_1 \lambda^{(2)}(2 + \lambda^{(2)} d^{(2)}) \end{aligned} \tag{68}$$

and the solution for $r_1^{(2)}$ is

$$r_1^{(2)} = \frac{(C_1+C_2)(1+\lambda^{(2)}d^{(2)}) \pm \sqrt{(C_1+C_2)^2(1+\lambda^{(2)}d^{(2)})^2 - 4C_1C_2\lambda^{(2)}d^{(2)}(2+\lambda^{(2)}d^{(2)})}}{2C_2d^{(2)}} \tag{69}$$

The valid solution is the one with the positive sign before the square root in the above equation because only in this case is the value of $r_1^{(2)}$ greater than $\lambda^{(2)} + \frac{1}{d^{(2)}}$. It can be seen that,

$$\frac{r_1^{(2)}}{r_2^{(2)}} = \frac{C_1}{C_2} \tag{70}$$

so the OPT policy allocates resources in the ratio of the available capacities.

Appendix C

We derive here the resource allocation for the adaptive resource allocation policy proposed in Section 3.3. The policy is defined as minimizing the function

$$S = \sum_{i=1}^{h} \frac{r_i}{C_i - A_i} \tag{71}$$

subject to the delay constraint

$$\sum_{i=1}^{h} \frac{1}{r_i - \lambda} = d \tag{72}$$

Using this equation, we can eliminate r_h from the equation for S, obtaining

$$S = \sum_{i=1}^{h-1} \frac{r_i}{C_i - A_i} + \frac{1}{C_h - A_h}(\lambda + \frac{1}{d - \sum_{j=1}^{h-1} \frac{1}{r_j - \lambda}}) \tag{73}$$

S is minimized when its gradient vanishes,

$$\nabla S \equiv (\frac{\partial S}{\partial r_1}, \frac{\partial S}{\partial r_2}, \cdots, \frac{\partial S}{\partial r_{h-1}}) = 0 \tag{74}$$

where the i^{th} partial derivative is

$$\frac{\partial S}{\partial r_i} = \frac{1}{C_i - A_i} - \frac{1}{C_h - A_h}\left(\frac{1}{(r_i - \lambda)(d - \sum_{j=1}^{h-1}\frac{1}{r_j-\lambda})}\right)^2 \quad i = 1,\ldots,h-1 \tag{75}$$

We note that $\sum_{j=1}^{h-1}\frac{1}{r_j-\lambda} = d - \frac{1}{r_h-\lambda}$. Substituting in the above equation and carrying out a few manipulations, we obtain

$$r_i - \lambda = f(i,h)(r_h - \lambda) \quad i = 1,\ldots,h \tag{76}$$

where $f()$ is defined by

$$f(i,j) \equiv \sqrt{\frac{C_i - A_i}{C_j - A_j}} \quad i,j = 1,\ldots,h \tag{77}$$

Substituting the relations between the r_i's in the delay constraint (72), we can solve for r_h and hence for the capacity allocation at node i,

$$r_i = \lambda + \frac{1}{d}\sum_{j=1}^{h} f(i,j) \quad i = 1,\ldots,h \tag{78}$$

Since $f(i,i) = 1$ and $f(i,j) > 0$, the r_i's satisfy the condition $r_i > \lambda + \frac{1}{d}$ and thus are valid solutions. Further, it can be easily shown that,

$$\frac{r_i - \lambda}{r_j - \lambda} = \sqrt{\frac{C_i - A_i}{C_j - A_j}} \quad \forall i,j \tag{79}$$

which is similar in spirit to the resource allocation ratio for the OPT policy of Appendix B.

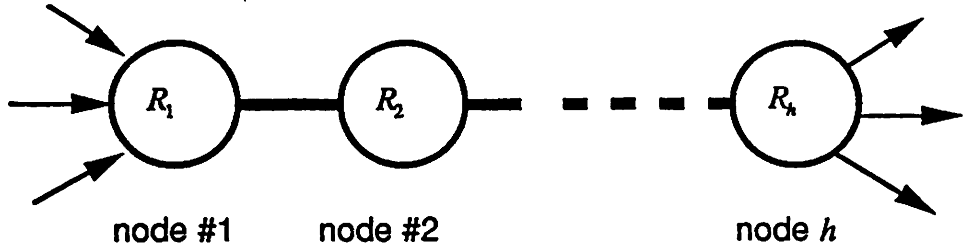

Figure 1: The network model.

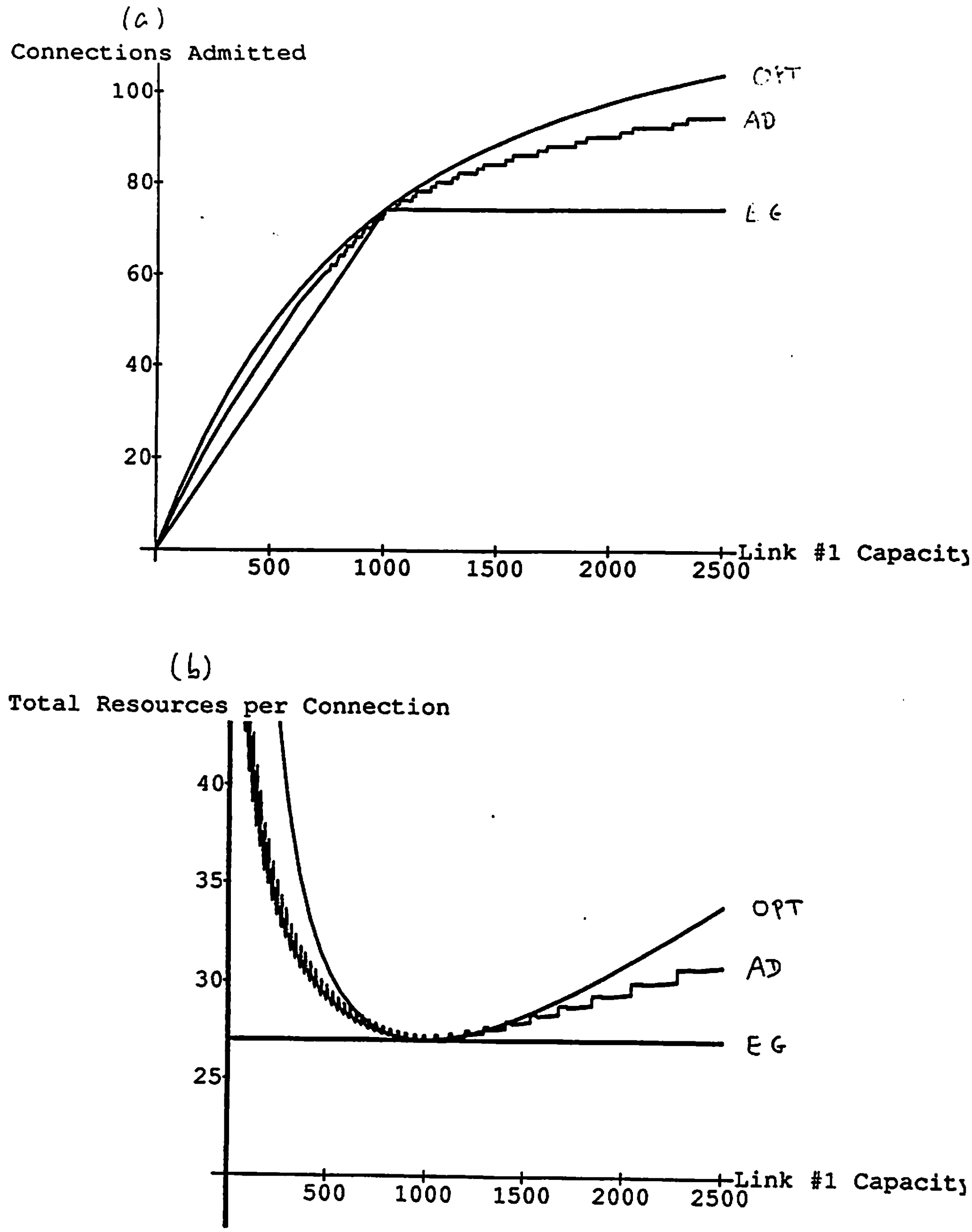

Figure 2: Results for the guaranteed-delay model with $h = 2$ and $C = 2$ as the capacity of link #1 (C_1) is varied. The policies shown are: the optimal (OPT), the adaptive heuristic (AD), and the equal-guarantee (EG). (a) Connections admitted. (b) Total resources per connection.

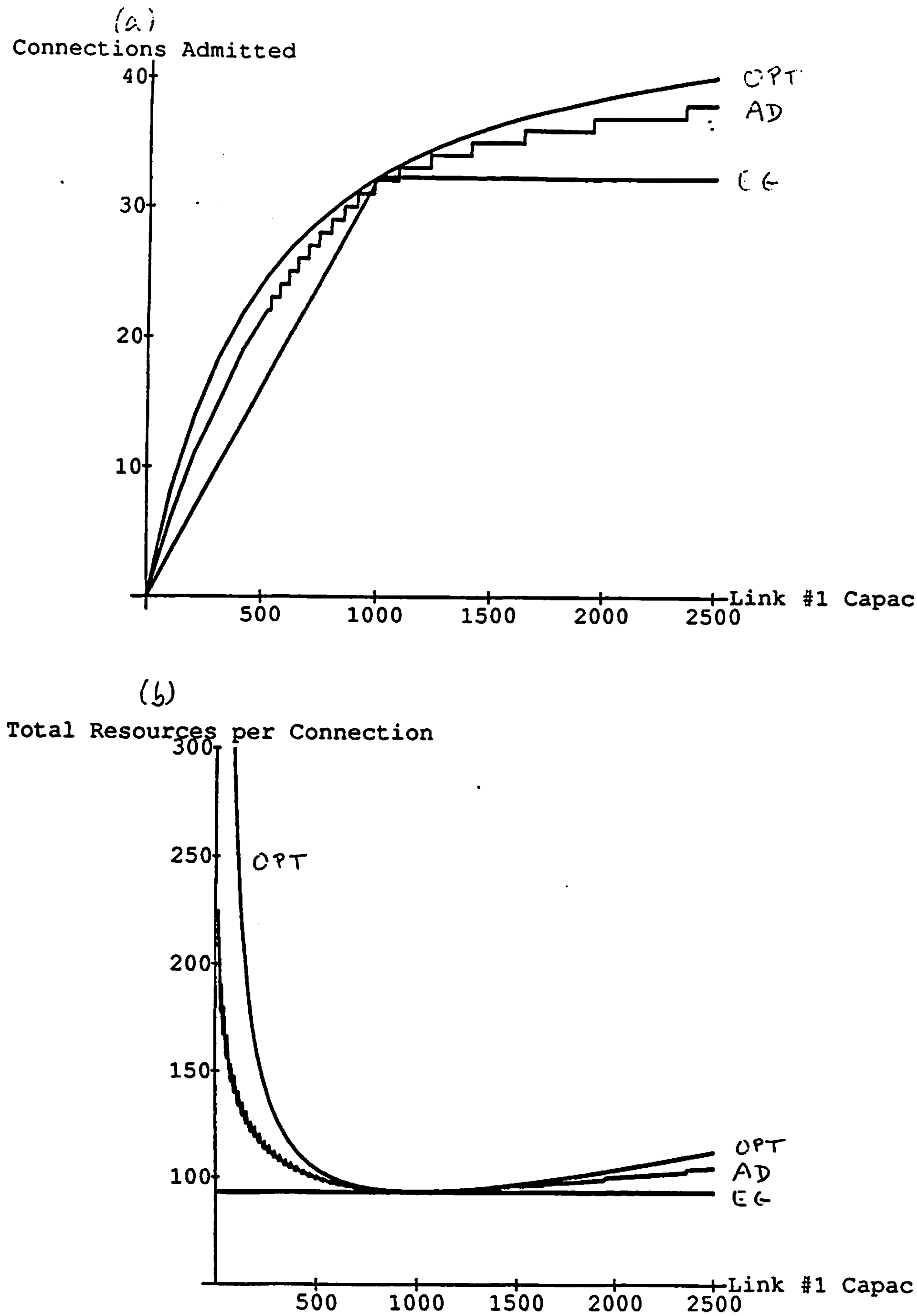

Figure 3: Results for the guaranteed-delay model with $h = 3$ and $C = 1$ as the capacity of link #1 (C_1) is varied. The policies shown are: the optimal (OPT), the adaptive heuristic (AD), and the equal-guarantee (EG).(a) Connections admitted. (b) Total resources per connection.

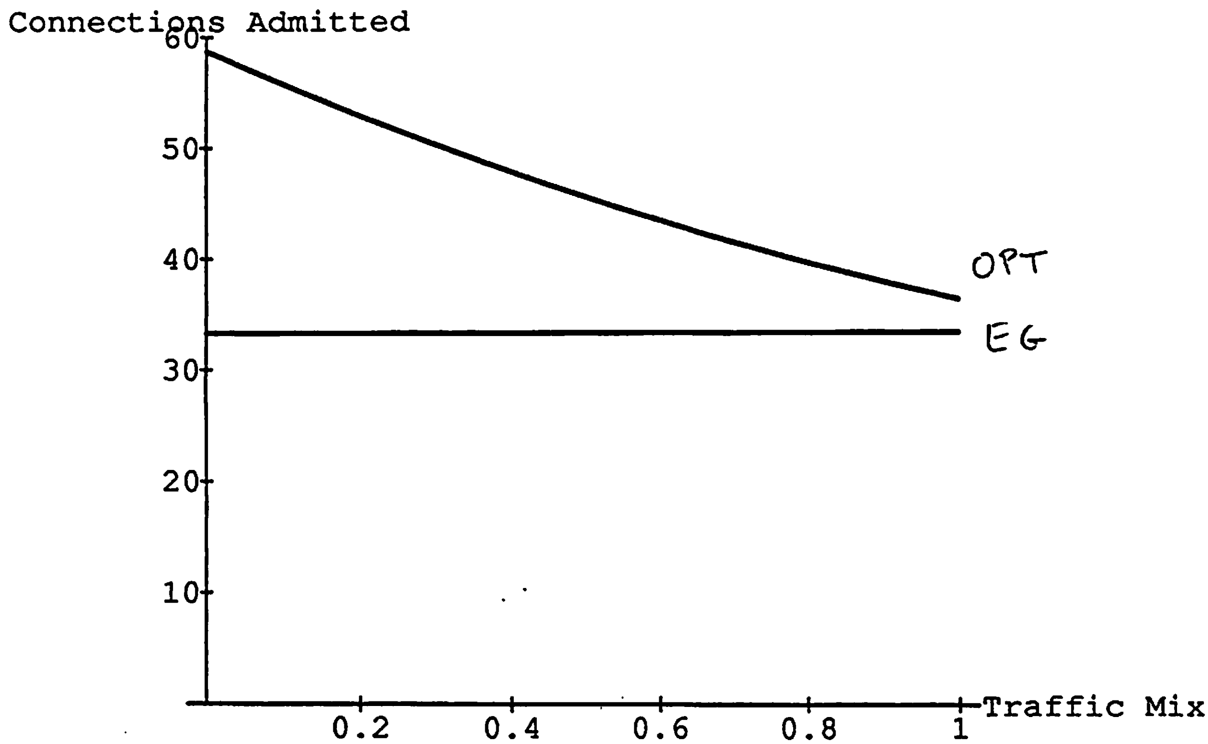

Figure 4: Connections admitted for the guaranteed-delay model with $h = 2$ and $C = 2$ as the traffic mix parameter (p) is varied.

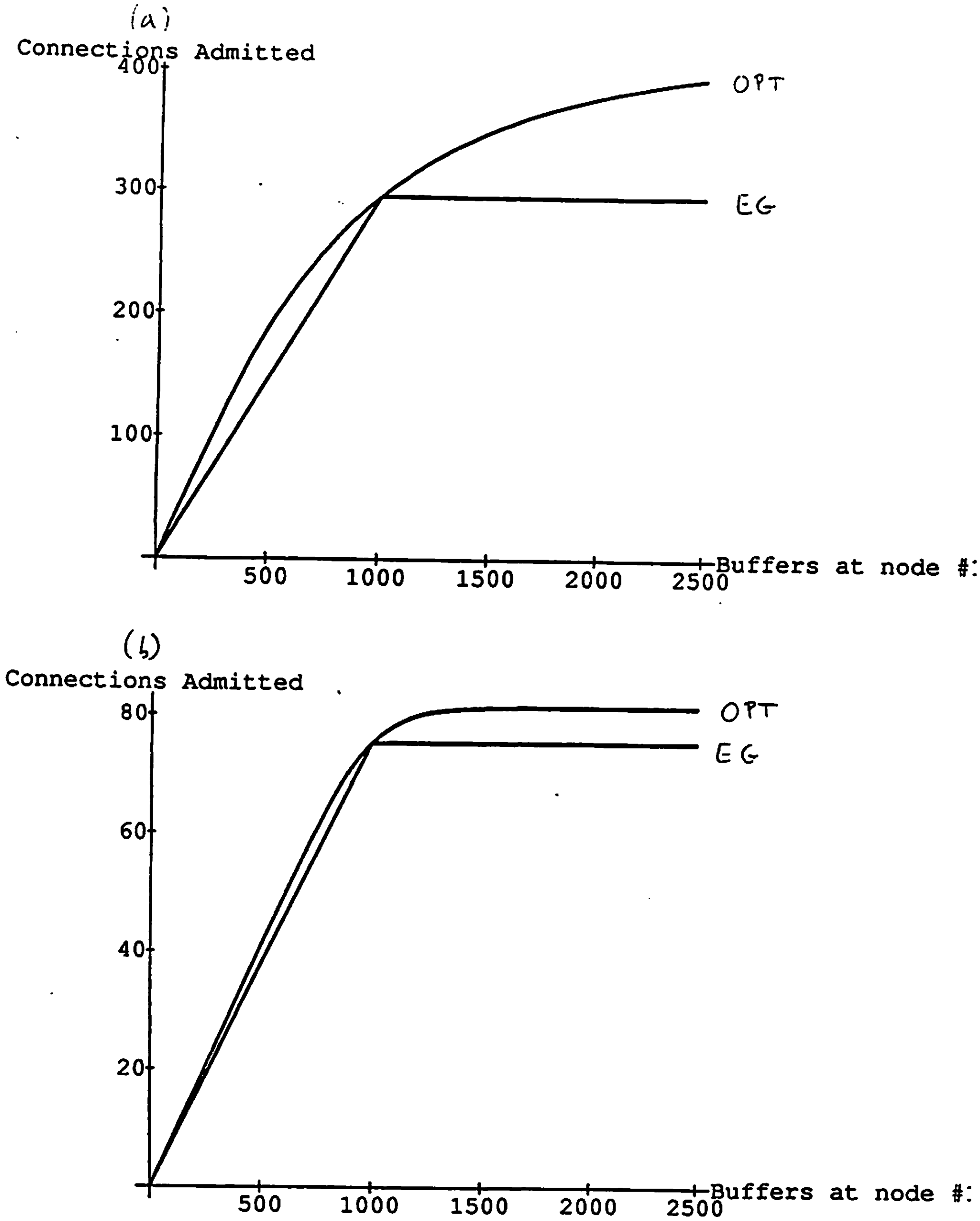

Figure 5: Connections admitted for the guaranteed-loss model as the size of the buffers at node #1 (B_1) is varied. (a) Relatively large guaranteed loss, $b = 0.1$. (b) Small loss, $b = 10^{-4}$.

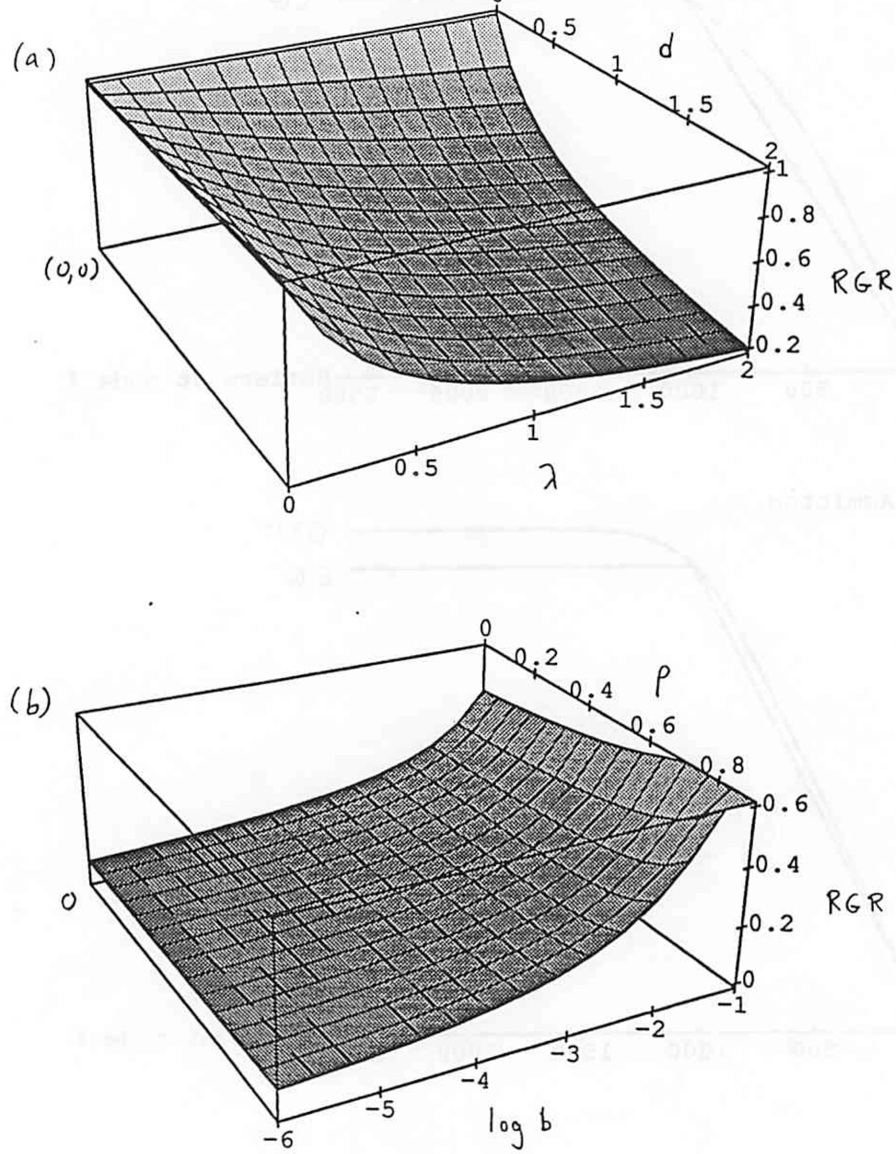

Figure 6: RGR against system parameters for: (a) The guaranteed-delay model. (b) The guaranteed-packet-loss model.