Routing and Admission Control Algorithms for Multimedia Traffic *

Sanjeev Rampal† and Douglas S. Reeves‡
Departments of Computer Science and Electrical & Computer Engineering
North Carolina State University
Raleigh, NC 27695 USA

Abstract

Interactive voice and video applications (e.g. teleconferencing) over multi-hop packet-switched networks require bounds on delay, loss, and jitter. This paper compares routing algorithms and admission control methods which are intended to provide the quality of service required by such applications. The method of comparison is detailed simulation. As part of this work, we propose several novel real-time routing algorithms.

The best-performing routing algorithm is a new method which takes into account the constraints imposed by the admission control algorithm. A routing algorithm similar to those used in the phone system is the worst of the dynamic algorithms; static routing is far worse than any of these. The differences between the routing algorithms are greatest when the network is highly connected, and when admission control constraints are difficult to meet. The issue of *fairness* is also considered in the evaluation of the routing algorithms.

Three admission control algorithms were compared, under varying traffic conditions and QoS requirements. The impact of traffic shaping on link utilizations was investigated. Overall, the choice of admission control method was more important than the choice of routing algorithm.

This work is the first detailed evaluation of routing algorithms for voice and video traffic in multi-hop networks that provide strong end-to-end guarantees.

1 Introduction

Packet-switched data networks are increasingly being utilized to carry multimedia traffic (e.g., video and voice). This trend is expected to continue with the deployment on a

*This work was supported by the Air Force Office of Scientific Research Under Grant F49620-92-J-0441.
†e-mail: sdrampal@eos.ncsu.edu
‡e-mail: reeves@csc.ncsu.edu

	Peak Rate	Mean Rate	Allowable Packet Loss
Voice	32 Kb/s	11.2 Kb/s	.05
Video	11.6 Mb/s	3.85 Mb/s	10**(-5)

(a) Source models for voice and video, and typical acceptable loss rates.

CCITT G.114 Delay Recommendations	
One-Way Delay	**Characterization of Quality**
0 to 150 ms	"acceptable for most user applications"
150 to 400 ms	"may impact some applications"
above 400 ms	"unacceptable for general network planning purposes"

(b) End-to-end delay requirements for voice and video.

Table 1: Typical data rates and QoS requirements for multimedia traffic.

widespread scale of high-speed networking technology, such as ATM. The new networking technology is intended to fully support both real-time and non-real-time traffic. Multimedia traffic has fairly predictable characteristics, and stringent quality-of-service (QoS) requirements for end-to-end delay, jitter, and allowable loss. Table 1 shows the predicted data rates and typical QoS requirements of voice and video [17][21][25].

The quality-of-service required by a multimedia application can only be guaranteed if network resources (bandwidth, buffer space, processing time) are reserved in advance. The end-to-end connection between a source and a destination carries a continuous stream of data. We will refer to one such connection as a *call*, or a *channel* (both terms are used in the literature). Before a connection can be established, a *call-level admission control* (CAC) procedure determines if sufficient resources exist in the network. If there are sufficient resources to guarantee that the requested QoS can be achieved, the resources are reserved and the call is accepted into the network; otherwise, the call is rejected, or *blocked*.

The provider of a network typically wants to maximize the profit from the network. For a fixed network cost and configuration, this means the goal is to accept as many calls as possible. Based upon an explicit tariff structure, certain types of calls may be favored in order to improve overall profitability. Maximizing the probability of acceptance is also important to the users of the network. In addition, users value the *fairness* of a network. This means either all classes of calls have equal likelihood of being accepted, or the tariff structure which establishes priority is simple and "reasonable".

The CAC method used by a network directly determines which channels are accepted, and which are blocked. Thus, admission control is of vital importance to both the network provider and the users. Another important function of a network is *routing*, which finds a path in the network from a source to a destination. In a connection-oriented service, all packets from a call follow this same path from the source to the destination. Routing itself does not accept or reject calls. However, the choice of route directly influences the likelihood

that a call will be accepted. As an example, if a routing method selects a route which has very few resources available, the request for a connection will probably be rejected. Once a route is selected, the availability of resources along other paths is of no use in call admission. As a result, routing is also important to the network provider and users.

Multimedia traffic has special QoS requirements which can only be met by admission control methods that are specifically constrained to provide those types of guarantees. This traffic is expected to be a major portion of the load in new high-speed networks. Thus, it is important to know how existing routing algorithms will perform on this type of traffic, and if they don't perform well, to devise algorithms which will work well. Little work has been done on this important topic. This previous work includes the following.

- Parris and Ferrari [20] proposed a routing algorithm for real-time applications (described in Section 3 as the SP algorithm) but did not investigate its performance.

- Kompella et al [16] have investigated some routing problems for multicasting of multimedia data. They show that finding an optimal multicast tree which meets delay bound constraints is an NP-complete problem, and present a heuristic approximation.

- Ahmadi et al.[1] investigated dynamic routing algorithms for the Paris networking project. They proposed a method which computed the shortest path (where distance was a function of link utilization) satisfying the delay bound, and having the minimum possible hop count. This algorithm was shown to be much better than a minimum-hop algorithm, and somewhat better than a pure shortest-path algorithm. The routing algorithm was evaluated only in combination with the admission control method used in the Paris project. This CAC method is based on statistical multiplexing and the computation of equivalent capacity.

Our goal in this paper is to examine the impact of routing algorithms on the blocking probability of real-time channels. We formulated and implemented several new routing algorithms to specifically address the QoS requirements and admission control constraints of multimedia traffic. We experimentally investigated (using simulation) the relative performance of the algorithms for a variety of admission control algorithms and traffic loads. This paper is the first in-depth investigation of this important topic. As a secondary issue, we also scrutinized the performance of several admission control policies for multimedia data. We restricted our scrutiny to deterministic policies which can provide strict *a priori* guarantees of QoS; no new policies are proposed herein. This represents the first detailed, quantitative comparison of these policies for a variety of traffic conditions. The experiments yielded a number of important insights into router and admission controller behavior.

The organization of the paper is as follows. In the next section we review the three admission control methods used in our experiments, and also describe the use of traffic shaping with those methods. In Section 3, we describe the routing algorithms which were evaluated. Section 4 describes our simulation method and assumptions, and Section 5 presents the results of our experiments. The last section summarizes our findings, and lists some open problems.

2 Admission Control Schemes for Real-time Traffic

In this section, we describe three well-known deterministic methods of admission control. We used these three methods to investigate the interaction between routing and admission control of multimedia traffic.

The CAC policy must first calculate the resource requirements for a call request, and then check whether the existing calls in the network leave enough resources available to accommodate the request. This can only be determined if the admission control algorithm knows how packets will be served at each switch along the path of the requesting call. As an example, QoS guarantees and resource requirements will be very different if the service discipline for packets is First-In-First-Out, or Round Robin, or Static Priority Order. In this paper, we will use the term "call admission" to refer to the service policy at the switches, as well as the resource reservation scheme; this is not standard usage, but is convenient for our purposes.

The resources required for a call are calculated from some characterization of the properties of that call. To make the calculation simple, only a few properties are used, such as the packet peak and mean arrival rates, and the maximum burst length.

The service policy determines how packets needing to use the same outgoing link will be multiplexed onto that link. The two major types of multiplexing are *statistical multiplexing* and *deterministic multiplexing*. Statistical multiplexing relies on assumptions about the lack of correlation between individual channels to make predictions about their aggregate behavior. Under such assumptions, very high link utilizations can be achieved, but the guarantees which can be given are very much dependent on the validity of the assumptions. Deterministic multiplexing makes no such assumptions about correlation, and as a result involves a worst-case analysis of the aggregate behavior of the multiplexed channels. Deterministic multiplexing produces strong guarantees, but is conservative and so normally results in lower link utilizations. A much more complete discussion of these issues in the context of real-time traffic, including a survey of existing work, can be found in reference [2].

In our study we used exclusively deterministic methods of admission control and multiplexing. The only effective way to calculate the QoS produced by statistical methods in multi-hop networks is to simulate the end-to-end transmission of every packet. This can require a huge amount of CPU time for high-speed networks. In addition, a fair comparison of policies requires all performance parameters except one to be fixed. For example, to meaningfully compare the delays resulting from two different admission control policies, we must be sure that the loss rates for the two policies are the same. However, there is no known way to accurately fix the end-to-end loss rate that a given statistical policy will produce.

By contrast, when a deterministic admission control method is used, the QoS can be calculated purely by simulating the admission control function. This requires several orders of magnitude less CPU time than packet-level simulation. One can also easily control the experiments to ensure that all admission control methods provide the same quality-of-service (loss and delay)[1]. Thus, the results from multiple experiments can be directly and fairly

[1]Our experiments did not place any restrictions on jitter. There may be significant differences between

compared.

The three admission control methods which we used in this study are Earliest-Due-Date (EDD)[7], Stop&Go[11], and Weighted-Fair-Queuing (WFQ)[19]. Each of these is described briefly below. The low utilization which is a drawback of deterministic methods can be improved through the use of traffic shaping, at the expense of some packet loss. We also describe a method of traffic shaping for multimedia traffic.

2.1 Earliest Due Date

The Earliest Due Date (EDD) packet scheduling and admission control policy [7], [15], splits end-to-end delay bounds into local per-link delay bounds. The sum of the link delay bounds along the path from the origin to the destination must be equal to the end-to-end delay bound. Packets for an outgoing link are multiplexed according to the Earliest Due Date policy, where the due date is calculated as the sum of the expected arrival time of the packet at the switch, and the local delay bound (or *deadline*) for the outgoing link.

Let the local delay bound of channel i on link n be denoted as $d_{i,n}$. Also denote the minimum packet interarrival time at the source for channel i as $T_{min,i}$, and the worst-case transmission time of one packet of channel i on link n as $X_{i,n}$. If the K channels using a link n are ordered by increasing value of their local delay bounds for this link (so that $d_{1,n} \leq d_{2,n} \leq \cdots \leq d_{K,n}$), then the following necessary constraint must be satisfied to accept channel i on link n:

$$T_{min,i} \geq \sum_{j=1}^{K} X_{j,n}, \qquad (i = 1, \cdots, K) \tag{1}$$

Let $X_{max,n}$ be the maximum packet transmission time of any existing channel using link n. The requesting channel can be accepting without violating the guarantees for existing channels on the link if and only if

$$d_{i,n} \geq \sum_{j=1}^{i} X_{j,n} + X_{max,n} \qquad (i = 1, \cdots, K) \tag{2}$$

This equation also represents the maximum queuing delay which a packet for channel i will experience on link n. The end-to-end delay bound for this channel is then $D_i = \sum_{n=1}^{H_i} d_{i,n}$, where H_i is the number of hops on the path from the origin to the destination of channel i.

The EDD policy is based on results from real-time scheduling theory. It is considered to be somewhat expensive to implement, but appears to be quite flexible.[2]

---

the admission control methods and routers with respect to jitter; this was not studied.

[2] We have described and run experiments with the "simple" form of EDD; the full version is even more flexible, and even more expensive to implement.

2.2 Stop&Go

The Stop&Go service and admission control policy [11] is designed to provide very tight bounds on end-to-end jitter in networks, as well as predictable end-to-end delays. Each requesting channel i is assigned to some *frame* of size (duration) T_g, and is allocated a transmission bandwidth r_i. During each interval of length T_g, i is allowed to transmit at most $r_i \cdot T_g$ bits; this is termed the (r_i, T_g)-smoothness property.

A link is assigned a set of G frame sizes, $T_1, \cdots, T_G$; assume the frame sizes are ordered in decreasing size. The total bandwidth of link n assigned to frame level g, denoted C_n^g, is the sum of the bandwidths (r_i's) for all channels assigned to frame level g. Let C_n be the total bandwidth of link n, and $l_{max,n}$ denote the maximum packet size over all channels using link n.

A requesting call can be accepted by this link, without violating the QoS guarantees given to existing calls using the link, only if the following necessary constraint is satisfied:

$$-C_n^{g0} + \sum_{g=g0}^{G} C_n^g (1 + \lceil T_{g0}/T_g \rceil) T_g / T_{g0} \leq \begin{cases} C_n - l_{max,n}/T_{g0}, & g0 = 2, \cdots, G \\ C_n, & g0 = 1 \end{cases} \quad (3)$$

When the ratios of the frame sizes are integers, instead of Equation 3 reduces to a simple capacity constraint [11].

There is an important source of bandwidth wastage called *quantization*, which can be explained in the following way. Many networks have a fixed or lower limit on the size of a packet; let this be denoted l_{unit}. The maximum number of packets of this size which will be transmitted by channel i during one frame is $\frac{r_i T_g}{l_{unit}} = x$. For transmission purposes, x must be an integer. If it is not, r_i must be rounded up, i.e., its new value is computed as $r_i = \lceil x \rceil \frac{l_{unit}}{T_g}$. The difference between the old (non-quantized) and new (quantized) value of r_i represents bandwidth that will be wasted due to quantization. When x is small and l_{unit} is large or T_g is small, this bandwidth wastage can be substantial.

For a channel i routed over a path with hop-count H_i, and assigned to a frame of size T_g at every link along the path, the maximum end-to-end-delay D_i it will experience is guaranteed to be

$$H_i \cdot T_g \leq D_i \leq 2H_i \cdot T_g \quad (4)$$

Stop&Go provides very predictable service, and is not particularly difficult to implement. It does require sources to obey the smoothness property. Also note that the end-to-end delay is related to (i.e., dependent upon) the rate allocated to the channel.

2.3 Weighted Fair Queuing

The Weighted Fair Queuing (WFQ) method was proposed by Demers et al.[5] and analyzed by Parekh[19]. We summarize a restatement of that analysis due to Partridge[21]. In WFQ, channels sharing an outgoing link are transmitted as if they were serviced in Round-Robin order on a bit-by-bit basis. Channels can receive differing amounts of bandwidth by using a Weighted Round-Robin service scheme ("weighted" so that some channels can transmit more

than 1 bit on each round). Tight bounds on end-to-end delay can be achieved if the arrival process for the channel at the source is regulated by a token bucket with bucket capacity β_i and token generation rate ρ_i. Due to restrictions on the length of this paper, we do not describe token buckets; a good introduction can be found in [21]. A channel i is assigned a transmission bandwidth or rate of $r_i > \rho_i$ on all links along its path.

A necessary constraint which must be satisfied in order to accept a new channel on a link n is that the sum of the rates of all the channels multiplexed onto this link be less than the capacity of the link. Thus, if K channels have been accepted on this link,

$$\sum_{i=1}^{K} r_i \leq C_n \tag{5}$$

The delay bound for a requesting channel is

$$D_i \leq \frac{\beta_i}{r_i} + (H_i - 1)\frac{l_i}{r_i} + \sum_{n=1}^{H_i} \frac{l_{max,n}}{C_n} \tag{6}$$

The rate assigned to a channel is a function both of the peak rate of the source (after regulation by the token bucket), and of the delay bound which is desired.

WFQ has received a great deal of attention lately, and is considered to be both reasonable to implement, and to provide high quality of service.

2.4 Traffic Shaping and Effective Bandwidths

The *effective bandwidth* (also referred to as *equivalent capacity*) of a source represents a minimal bandwidth that must be provided to guarantee its required QoS [6][13]. For this purpose, QoS is defined as a function of delay and allowable loss rate. The computation of effective bandwidth is possible for sources which conform to certain bounds on arrivals; see the cited references for details. The enforcement of the effective bandwidth is achieved by *traffic shaping*. A traffic shaper delays the arrival of some packets in order to smooth the arrival process for the channel. This delay can be accomplished by mechanisms such as leaky and token buckets (again, see [21] for an introduction). If the buffer capacity is exceeded (input rate exceeds output rate for too long a period), some packets may be discarded. The token generation rate and bucket size are calculated to ensure the losses do not exceed the user-specified quality-of-service bound.

The benefit of traffic shaping is that it can improve network utilization while keeping delay and loss bounds within acceptable levels. Traffic shaping of a source can be combined with a deterministic admission control policy to yield strong end-to-end quality-of-service guarantees, with improved network utilization; see Figure 1. The traffic shaper introduces an additional delay which must be subtracted from the specified end-to-end delay bound to yield the allowable network delay. When the network delay bound is considerably less than the required end-to-end delay bound, traffic shaping can significantly reduce the effective bandwidth. Since the deterministic admission control methods guarantee zero loss inside the network, all allowable loss occurs at the network interface. The effective bandwidth of a channel after shaping is regarded as the peak bandwidth of the channel, for purposes of admission control.

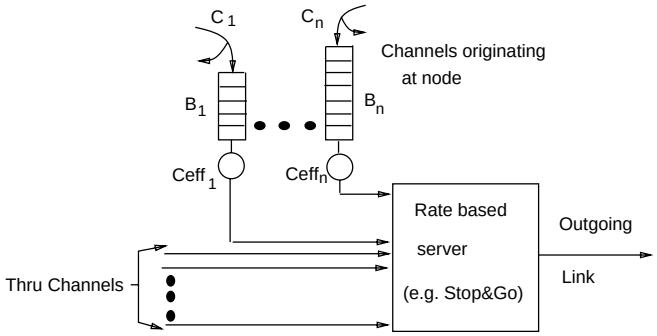

Figure 1: Use of traffic shaping with a rate-based server, for a single output link of a switching node.

3 Routing methods

The routing problem has been investigated extensively for both circuit-switched and packet-switched networks. Some recent summaries of routing techniques are by Bertsekas [3], Girard [10], and Perlman [22]. The goals for routing include maximizing the load accepted by the network while providing satisfactory QoS to channels and treating all call requests equally (i.e., being fair). These goals can be contradictory, so the search for optimal algorithms is in fact quite elusive. In addition, there are many implementation factors which can significantly complicate this search. For instance, a hierarchical routing method may be needed for networks and internetworks with large numbers of hosts.

Routing methods can be static or dynamic. Static methods are extremely simple to implement, but are unable to react to changing network traffic patterns, This can easily lead to unnecessary and very localized congestion. Dynamic methods attempt to balance the load across the network and thus accept more traffic with less delay variation across paths. Dynamic routers often optimize some metric (such as expected delay) subject to certain constraints on the choice of links. Dynamic methods are more complex than static methods, and are also subject to problems of oscillation. For details, see any of the surveys cited above.

Most dynamic algorithms are based upon the concept of a shortest path. These are usually computed by Djikstra's algorithm or the Bellman-Ford algorithm. Considerably different routing goals can be accomplished by suitably defining the *length* of a link, while the algorithm executed remains exactly the same.

Little work has been done on routing of multimedia traffic with delay and loss requirements, as mentioned in Section 1. We implemented a wide variety of routing algorithms to determine what effect routing has on call acceptance. These algorithms can be roughly classified into three groups:

- "Conventional" algorithms which have been considered for packet-switched networks;
- Sequential algorithms which have been proposed for use in circuit-switched networks; and,

- Real-time algorithms which promise to better fulfill the quality-of-service requirements of multimedia traffic.

We now describe each of the routing algorithms.

3.1 Conventional Algorithms

As a point of comparison, we chose two simple, generic routing algorithms which have been used in packet-switched networks.

The **C_STAT** algorithm is a static router. For each origin-destination pair, a minimum-hop path is found. If there are multiple minimum-hop paths between the origin and destination, the one which uses the lowest numbered link (according to a static link ordering) possible at each hop is chosen. The selected path for each destination is stored for each host. All calls originating at the same origin and having the same destination follow the same path, regardless of network loading conditions.

The **C_DYN** algorithm is a dynamic shortest-path router. The length of each link is set to be inversely proportional to the spare capacity fraction (i.e., 1 - utilization) of that link. The implicit goal of C_DYN is to minimize the end-to-end delay encountered by a channel. A desirable by-product is that the load on the network tends to get evenly distributed.

3.2 An Algorithm Based on Circuit-Switched Network Routing

Hwang et al [14] proposed the use of sequential routing algorithms, similar to those used in circuit-switched networks, for routing in integrated services networks. Because channels have many of the characteristics of circuits in circuit-switched networks, it seems reasonable to expect that circuit-switched routing algorithms would perform well. Elwalid [6] also suggested that this type of routing algorithm should perform well for real-time traffic when it is statistically multiplexed based on the concept of effective bandwidths. These algorithms also have the advantage that they are totally distributed, and do not require updates of routing tables.

The best-performing member of this group is known as the Crankback **(CB)** algorithm. In the Crankback algorithm, a path from the origin to the destination is found one hop at a time. From the origin, the algorithm checks the first link on the static minimum hop path to the destination. If the call can be accepted on that link (according to the admission control constraint), that link is added to the path. Otherwise, each of the remaining links from the origin are tried, in (static) link order. The first one for which the call can be accepted is added to the path. Each time a link is added to the path, routing continues recursively from the node at the other end of the link. The algorithm terminates if the destination is reached (success). If a node is encountered for which no outgoing links can accept the requesting call, backtracking to the previous node on the partially-routed path is allowed. A different outgoing link is chosen from this node (again, in static order) to avoid the congested node. If this process exhausts all paths starting from the origin without successfully reaching the destination, failure to find a route is reported. Once a feasible path is found, the call is accepted if it will meet its end-to-end-delay bound along this path.

3.3 Real-Time Algorithms

None of the above algorithms is formulated to specifically meet the needs of multimedia traffic. We use the term *real-time routing algorithm* for any method which considers the delay bounds and necessary constraints of the call admission function. Several such methods are presented below; except where noted, they are proposed here for the first time.

These algorithms are constrained to avoid links which will certainly lead to call rejection. Any link for which the necessary constraint of the admission control algorithm (Equations 1 for EDD, 3 for Stop&Go, and 5 for WFQ) is not fulfilled is eliminated from consideration by the routing algorithm. That is, the network configuration is pruned before the routing algorithm is run.

The Shortest Path **(SP)** algorithm is targeted to find a path which minimizes the delay bound which can be guaranteed to the requesting channel. Let $d_{i,n}^{min}$ denote the minimum delay bound which can be guaranteed for a requesting channel i on link n, without invalidating the guarantees for existing channels which use that link. $d_{i,n}^{min}$ is calculated as follows:

- EDD: The smallest value of $d_{i,n}$ is computed for which the admission control condition (Equation 2) is still true. Since deadlines of channels already accepted on the link are already ordered by value, finding this smallest value is not difficult.

- Stop&Go: Let $T_{g_{min}}$ be the smallest frame for which the quantized bandwidth r_i can be accommodated, according to Equation 3. $T_{g_{min}}$ is not difficult to find, since in a real system there will likely only be a small number of frame sizes. $d_{i,n}^{min}$ is equal to $2T_{g_{min}}$.

- WFQ: The minimum value of $r_i \geq \rho_i$ which satisfies Equation 6 can be computed analytically. Given this value for r_i, the maximum delay for a packet of channel i on link n will be $\frac{l_i}{r_i} + \frac{l_{max,n}}{C_n}$.

The length of each link n is set to $d_{i,n}^{min}$ and then a shortest-path calculation is performed. The advantage of the SP algorithm is that worst-case delays are calculated directly from the admission control policy, rather than indirectly from the link utilization. The SP algorithm was first proposed in [20]. However, no quantitative evaluation was presented.

The Shortest Cost **(SC)** routing algorithm is targeted to meet server constraints, rather than minimizing end-to-end delay. This approach seems reasonable in the case where end-to-end delay bounds are not particularly difficult to achieve. In such a case, the server constraint is of more importance than the delay bound for a link. The length of link n is modeled in the following way for each admission control method:

- EDD: From Equation 1 it is clear that the probability of call i being blocked on link n increases as the value of the r.h.s approaches that of the l.h.s.. Acceptance should be improved when this difference is as large as possible. Hence, the length of link n is set to $1/(T_{min,n} - \sum_{j=1}^{K} X_{j,n})$, where $T_{min,n}$ is the minimum of peak packet interarrival times over all channels using link n (including the requesting channel).

- Stop&Go: From Equation 3, it is seen that the blocking probability increases as the quantity on the l.h.s. approaches that on the r.h.s. i.e. the link capacity. The length of link n is set to $1/(C_n - (-C_n^{g0} + \sum_{g=g0}^{G} C_n^g(1 + \lceil T_{g0}/T_g \rceil)T_g/T_{g0}))$.

- WFQ: The necessary constraint for WFQ is just a bandwidth constraint. Blocking increases when utilization increases. Thus, the length of link n is set to $1/(C_n - \sum_{i=1}^{K} r_i)$.

The SC algorithm performs a shortest path calculation using one of the above definitions of the link length. No consideration is given to meeting the end-to-end delay bound.

We also implemented a minimum-hop version of algorithm SC, called the Modified Shortest Cost **(M-SC)** algorithm. This algorithm finds the shortest path (with link lengths as described above for the SC algorithm) from among all feasible minimum-hop paths. We implemented M-SC as an approximation of the algorithm of Ahmadi et.al. [1], in an attempt to compare their routing algorithm with others.

Another promising idea for routing is to identify a path with minimum total length (where link lengths are as described for the SC algorithm), as well as minimum total delay. Unfortunately this problem is NP-complete [8]. The Min Max Cost with Delay Bound **(MMCDB)** algorithm is a heuristic approximation to this ideal algorithm. A path is found for which the maximum length of any link in the path is minimized, while also satisfying the end-to-end delay bound. A sketch of the algorithm is as follows. The links are first ordered by length; the ith shortest link is renamed link i. Using a dynamic programming approach, on the i'th iteration the shortest path from the source to the destination is found which uses only links numbered 1 through i. This path is checked to see whether it will meet the end-to-end delay bound according to the admission control method (Equations 2, 4, and 6 for EDD, Stop&Go, and WFQ, respectively). The algorithm terminates the first time a path is found which satisfies the delay bound (success), or when no path is found after the Lth iteration (failure), where L denotes the number of links in the network.

3.4 Optimal Routing

One approach to routing which we did *not* investigate is optimal routing using either the flow-deviation or gradient projection techniques. [3] While it is undoubtedly useful to compare heuristics to an optimal method, this method is not suited for high-speed networks for the following reasons:

- There are no constraints on packet delays or losses; only average delay is minimized.

- Packets in a single call can be routed along different paths, which is inconsistent with the ATM standard.

- This method requires all calls be routed at the same time, in a batch. Old calls may have to be rerouted when new calls arrive. This seems to be an expensive proposition.

To our knowledge, no optimal routing method which includes delay and loss constraints and routes all packets of a call along the same path has been proposed.

3.5 Summary

The eight routing algorithms described in this section represent a wide range of approaches that might be used to route multimedia data. The static algorithm is clearly the least expensive to implement, because it is run one time only. All other algorithms are executed once for each call that requests admission. Most of these algorithms involve a shortest path computation, which can be executed in $O((L+N)*\lg N)$ time, where L denotes the number of links in the network, and N denotes the number of nodes. The computation for algorithm MMCDB takes $O((L+N)*N)$ time in the worst case. Since CB has the potential to try all paths in the network that start from the source node, it could be very expensive (factorial in the number of links!) to execute in the case of a highly-connected network. However, it has been found practical for use on real telephone networks.

Any of the above algorithms should be practical to implement (in terms of running time) for networks with no more than a few thousand nodes. We note that a multimedia call will frequently last for quite some time; for example, a videoconference may last for an hour or more. The time to find a good route with any of these algorithms is insignificant relative to this call duration. For truly huge networks, or when the routing and call setup time must be minimized, a static or hierarchical routing approach will be preferable (even though the quality of the results will be lower). Another possible time savings is to perform call setup at the same time as the route is being selected; the CB algorithm is an example of such an approach.

Having described the algorithms, we now discuss how they were evaluated.

4 Experimental Method

We ran a series of experiments to determine the "goodness" of the above routing algorithms for multimedia traffic with stringent quality-of-service requirements. This allowed us to model the network much more accurately than analytical models would permit, at the expense of requiring much more CPU time to compute. The simulations were performed in a static environment with the assumption of infinite call holding times and negligible call processing times. An important point to make is that only the call admission process was simulated. Because the loss and delay characteristics of deterministic methods are completely predictable, it was not necessary to simulate the transmission of individual packets. In all cases confidence intervals were calculated for the 95% confidence level, after running the same experiment 20 times using different numbers output by the same random number generator. These confidence intervals are noted in each graph and table.

The figure of merit used to evaluate the routing algorithms was the call acceptance probability. As noted in the introduction, the service provider is interested in "profit", which is proportional to (although perhaps not equal to) the number of calls accepted. Maximizing the acceptance probability is also important to network users. In the experiments, the acceptance probability was calculated as the (number of calls accepted) / (number of calls requested) over the last R requests; this gives a probability at one discrete point, rather than a cumulative probability over all calls. In our experiments, R was set to 300. This

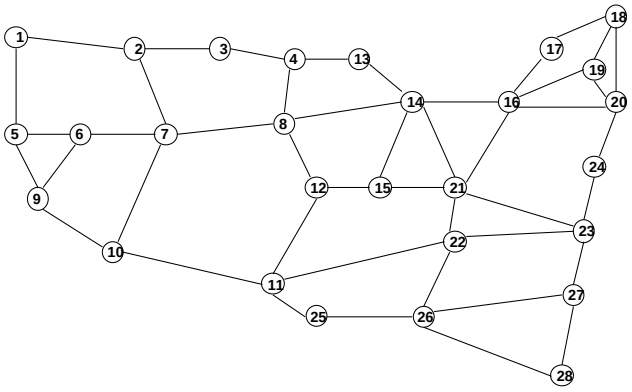

Figure 2: Graph of network used in the experiments (from [12]).

acceptance probability is measured as a function of the load in the network, which is the accepted load rather than the offered (sum of accepted plus rejected) load. Load is measured as the sum of the peak bandwidths of accepted channels.

We note there are a number of reasons why profit may not be simply a function of the number of calls accepted. For instance, calls requiring only one or two links may generate a higher profit (and may be preferred) to calls requiring more links. Another plausible example is that calls requiring more bandwidth may generate more profit than calls requiring less. We ran several experiments (not shown) using a figure of merit which assigned a higher "value" to calls needing a longer path and a higher bandwidth. The relative performance of the routers was unchanged for this alternative figure of merit, which gives us some confidence in the use of call acceptance probability for comparison purposes.

A good routing algorithm should also be fair, as well as produce high utilizations. Users will not be satisfied with a policy that consistently favors certain classes of traffic, unless they can be convinced that such a policy is reasonable. A further discussion of this subject may be found in [9]. We will examine both fairness and utilization in the analysis below of the experimental results.

Figure 2 shows the topology of the network used in the experiments. This network was previously used in work by Murakami [18] and Grover [12], and is loosely based on the Internet backbone in the U.S. The diameter of this network is 7 and the average node degree is 3.2. Link bandwidths were set uniformly to 155 Mb/s, except for one experiment noted below. The sum of propagation and processing delay at each link was chosen to be uniformly 3ms. We believe our assumptions for topology, delays, and bandwidths are realistic, based on recent trends.[3] Buffer sizes were assumed to be infinite, so no packet loss occurred inside the network.

Packet sizes were uniformly set to the ATM cell size of 53 bytes. Two standard Markov-modulated fluid models were used to generate voice and video traffic (from [25] and [17] respectively). Figure 3 shows the model parameters used in the experiments. These models

[3]In an earlier summary of this work [23], a smaller, slower, and more highly-connected network was used for the experiments. In the few cases where the difference in networks led to different conclusions, we note the differences with the earlier paper.

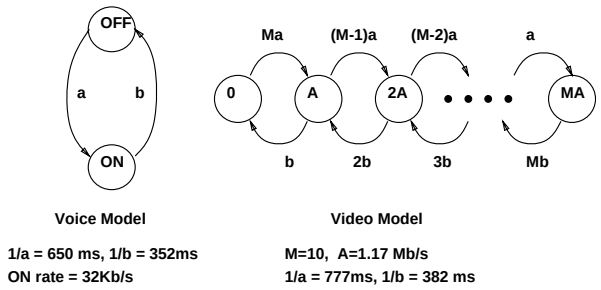

Figure 3: Voice and video models used in the experiments.

translate to a peak rate of 32 Kb/s for voice, and 11.7 Mb/s for video. The peak packet interarrival times (assuming 48 payload bytes per packet) were 12 ms for voice, and 33 μs for video. Note that a small amount of variability (on the order of 5%) was introduced into the source peak rates to more accurately reflect real behavior. That is to say, peak interarrival times for voice calls were evenly distributed between approximately 11.5 and 12.5 ms, while peak interarrival times for video were evenly distributed between approximately 32 and 34 μs. Allowable packet loss was as specified in Table 1. Allowable end-to-end delay was assumed to be 350 ms, with the exception of one experiment described below. The higher delay figure is acceptable according to subjective studies, while a much shorter delay is required to eliminate the need for echo cancellation equipment, or where a more natural interaction is preferred.

Several details about our implementation of the admission control methods should be mentioned, as follows:

- For the EDD algorithm, the deadline on a link n was set to be a_n times the end-to-end deadline (i.e., 350ms except for one experiment). a_n was computed as the fraction of the minimum achievable end-to-end delay which was due to link n.

- No method of choosing frame sizes or allocating channels to frame sizes has been published for Stop&Go. We chose frame sizes to be 3ms, 6ms, 12ms, 24ms, 48ms, 96ms, and 192ms. This avoids bandwidth loss due to using frame sizes that don't have common divisors.

- For routing purposes, a channel was assigned to the maximum possible frame size under which its end-to-end delay requirement could be met, under the assumption that the channel would be routed on a minimum-hop path to its destination.

- For WFQ, the channel rate r_i was set equal to the peak arrival rate for the channel. There was no attempt to discretize the channel rates into a set of allowable values for WFQ.

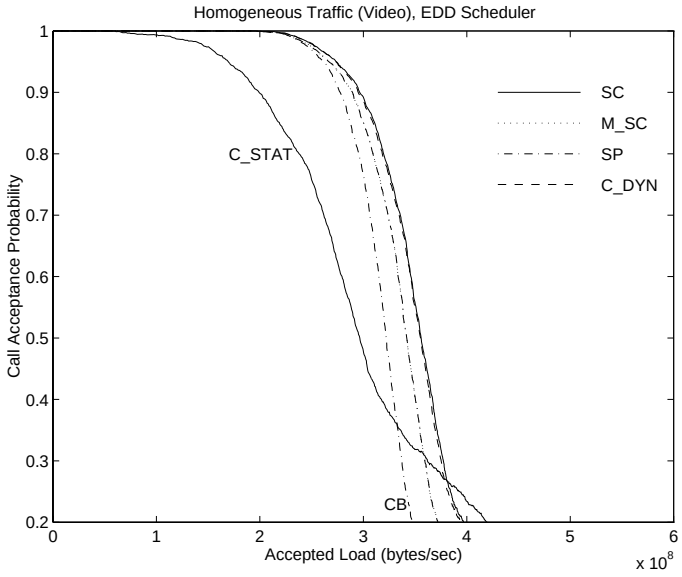

Figure 4: Call acceptance probabilities for homogeneous traffic (video only), EDD server, 155 Mb/s link speeds. 95% confidence intervals for any curve are no greater than 2.7%.

5 Experimental Results and Analysis

In this section we present the results of our experiments. The performance of the routing algorithms is investigated under different admission control policies and mixes of traffic. Note that a channel request can be blocked for any of the following reasons:

1. Bandwidth Rejection At least one link along the path does not have sufficient bandwidth available to accommodate the peak arrival rate of the channel.

2. Server Constraint Rejection For at least one link in the path, the necessary constraint imposed by the CAC function is not met.

3. Delay Bound Rejection The delay bound which can be achieved along this path violates the user-specified delay bounds for this channel.

5.1 Experiments Involving Homogeneous Traffic

As a base case for evaluating routers, we investigated the use of the EDD admission control policy for a network carrying only one kind of traffic (video). Figure 4 is a graph of the probability of call acceptance as a function of the accepted load in the network. Additional information about the performance of the routers is shown in Table 2.

For any acceptance probability above 60% (which is already lower than most users would consider tolerable), there is only a modest difference between the best and worst dynamic algorithms, while the static algorithm performs significantly worse, as expected. The best performing algorithm is the real-time algorithm SC, followed very closely by the C_DYN

	MMCDB	CB	SP	M_SC	SC	C_DYN	C_STAT
Path Lengths	3.34	3.06	2.83	2.72	2.76	2.47	2.22
Length/Min Hop	1.37	1.23	1.13	1.10	1.11	1.04	1.00
Link Peak Util.	.692	.668	.669	.656	.668	.593	.422

Table 2: Average path lengths of accepted channels, average ratio of path length to minimum-hop path length, and average link utilizations based on peak arrival rates. For homogeneous traffic (video only), EDD server, 155Mb/s link speeds. Measured at the 75% acceptance probability for each routing algorithm. Confidence intervals are no more than 1% for any number in this table.

algorithm. The MMCDB algorithm is not plotted because its performance in this and other experiments was consistently the worst of the real-time algorithms. The reason can be seen in Table 2; the ratio of path length to minimum-hop path length is higher, and as a result link utilizations are higher. The CB algorithm, in this and other experiments, performs consistently worse than the real-time algorithms, outperforming only the C_STAT algorithm. The CB algorithm gains implementation robustness/simplicity by taking a much more local view of path optimization, and as a result does not perform particularly well. Note from the table that the C_DYN algorithm is almost a dynamic minimum-hop router, and as a result performs very well when delay bounds are easy to meet and the traffic is uniform.

An interesting phenomenon in this and other experiments is that at some point the conventional static algorithm appears to outperform the best of the real-time algorithms(!). This conclusion is somewhat misguided for the following reason. Let $H_{i,j}$ be the number of hops on any minimum hop path between origin i and destination j. Table 2 shows that the conventional algorithms favor origin-destination pairs with significantly lower values of $H_{i,j}$. That is, channels which require a longer path (regardless of the routing algorithm) are being rejected at a significantly higher rate than channels which require shorter paths. When this behavior is continued for a long enough time, more channels will be accepted, since fewer resources are used for shorter paths (as shown clearly by the link utilization figures). Our conclusion is that the conventional algorithms (particularly C_STAT) are not as *fair* as the other algorithms.

Another illustration of this phenomenon is shown in Figure 5. A routing algorithm which only generates one-hop paths, and fails to find a path if the origin/destination nodes are not adjacent, is manifestly unfair to a large class of requesting calls. However, it clearly uses the minimum resources possible for each channel that is admitted, and at high network loads will surpass most other algorithms by our measure. This same observation has been made in [1]. To the categories of fairness identified in [9], we would add one more that is appropriate for high-speed networks: different types of calls should have roughly equal probability of acceptance.

The experiment shown in Figure 4 was repeated two times, once substituting the Stop&Go admission control policy for EDD, and once substituting the WFQ admission control pol-

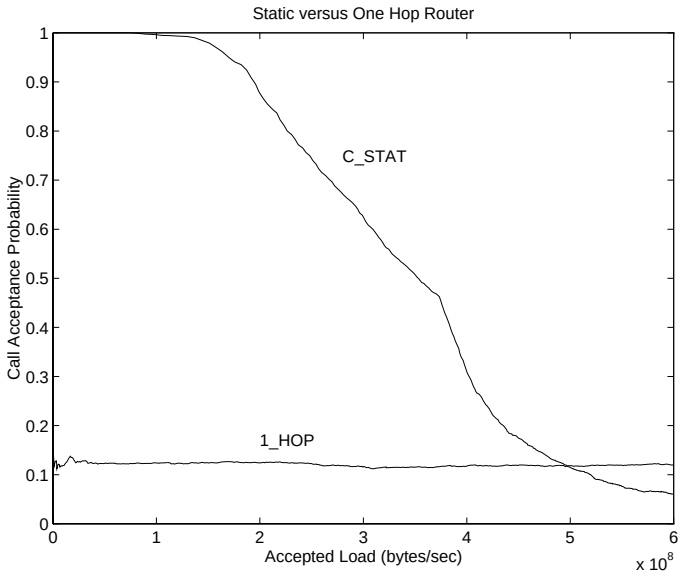

Figure 5: Call acceptance probabilities for homogeneous traffic (voice only), EDD server, 155 Mb/s links. 95% confidence intervals for any curve are no greater than 1.7%.

icy for EDD. The graphs were almost identical to those for EDD, and are not shown. For this combination of traffic type + QoS requirements + network bandwidth, the choice of admission control policy makes little difference.

5.2 Experiments Involving Heterogeneous Traffic

The previous experiments indicate a modest benefit from the use of a real-time routing algorithm. This is consistent across the three admission control policies, as long as the traffic mix is very homogeneous. The power of real-time scheduling theory is the ability to discriminate between tasks or messages with differing timing requirements. To explore the effect of a traffic mix with widely varying requirements, we ran experiments in which 80% of the traffic was voice, and 20% of the traffic was video. To emphasize the difference in these classes of traffic, we note that the peak rate of video is 365$\times$ the peak rate of voice.

Figure 6 shows the comparison of the routing algorithms when the admission control policy is EDD. Acceptance of a single video channel has a large impact on link utilization. Since packet sizes are assumed to be equal for all channels, the necessary constraint of EDD (Equation 1) dictates that no more than $(155 \times 10^6)/(11.7 \times 10^6) \approx 13$ channels can be accepted on any link for which at least one video channel is accepted. A link over which one video channel and only 12 voice channels are routed will obviously suffer a tremendous waste of bandwidth.[4] Under these more stringent traffic conditions the advantages of the real-time router SC is much more pronounced. A real-time routing algorithm such as SC

[4]To be fair, we note again that we have implemented the "simpler" version of EDD[7]. The full version would not suffer such drastic waste of bandwidth for this condition. However, it imposes a constraint on the channels which is much more expensive to compute.

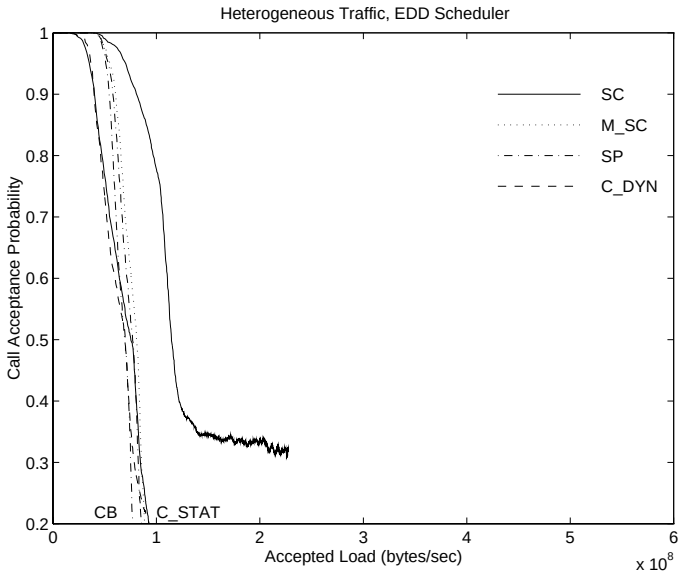

Figure 6: Call acceptance probabilities for heterogeneous traffic (20% video, 80% voice), EDD server, 155 Mb/s link speeds. 95% confidence intervals for any curve are no greater than 5.3%.

can provide substantially better network utilization when the particular constraints of the admission control algorithm are so important (i.e., are difficult to satisfy). Note that SC is also much better than the other previously-proposed real-time routing algorithm, the M_SC algorithm.

Figure 7 shows the results of an experiment for the same network and traffic conditions, but using the Stop&Go admission control policy. The real-time routing algorithms perform in the same order as for the case of homogeneous traffic: SC best by a considerable margin, then M_SC and SP (indistinguishable from each other), and CB worst. Both of the conventional algorithms (C_DYN and C_STAT) cross over the real-time algorithms at higher loads, which we attribute to the difference in fairness. The loads accepted by the Stop&Go policy are several times that accepted by the EDD policy. As mentioned, the necessary constraint of EDD (Equation 1) can lead to severe bandwidth waste for the heterogeneous traffic conditions that were simulated. There is no such waste for Stop&Go under these conditions. The Stop&Go necessary constraint (Equation 3) is almost a check that the sum of the peak bandwidths does not exceed the link bandwidth, under these conditions.

This experiment was run again using the WFQ admission control policy. The results are shown in Figure 8. While the relative rank of the routers for this experiment remains the same, the shape of the curves is somewhat peculiar. To understand this shape, refer to Figure 9. In this figure only two of the routing algorithms are plotted. The acceptance ratios of the two classes of traffic (voice and video) are graphed separately, as well as the combined total overall. Video calls are rejected while network utilization is still relatively low; significant rejections of voice calls occurs at a much higher load. When these two classes of traffic are combined, the resulting acceptance exhibits the expected first drop

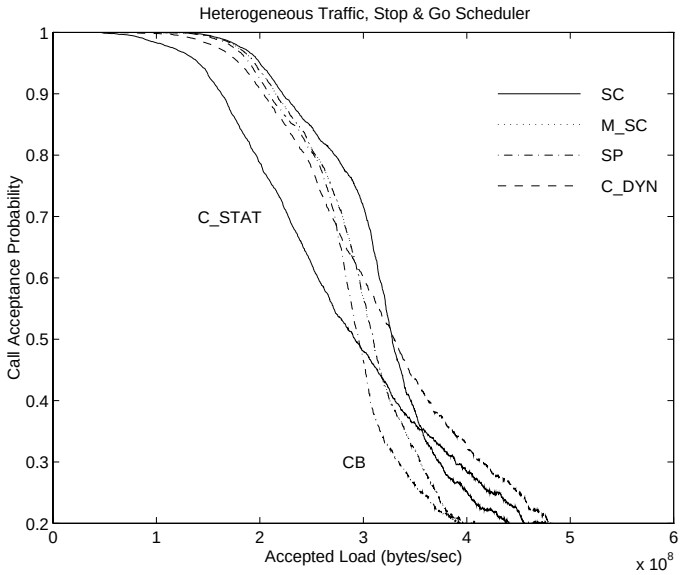

Figure 7: Call acceptance probabilities for heterogeneous traffic (20% video, 80% voice), Stop&Go server, 155 Mb/s link speeds. 95% confidence intervals for any curve are no greater than 7.6%.

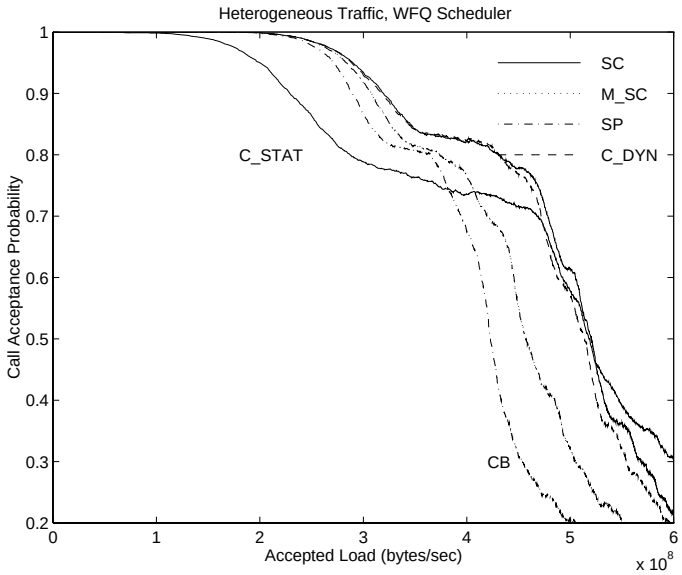

Figure 8: Call acceptance probabilities for heterogeneous traffic (20% video, 80% voice), WFQ server, 155 Mb/s link speeds. 95% confidence intervals for any curve are no greater than 8.9%.

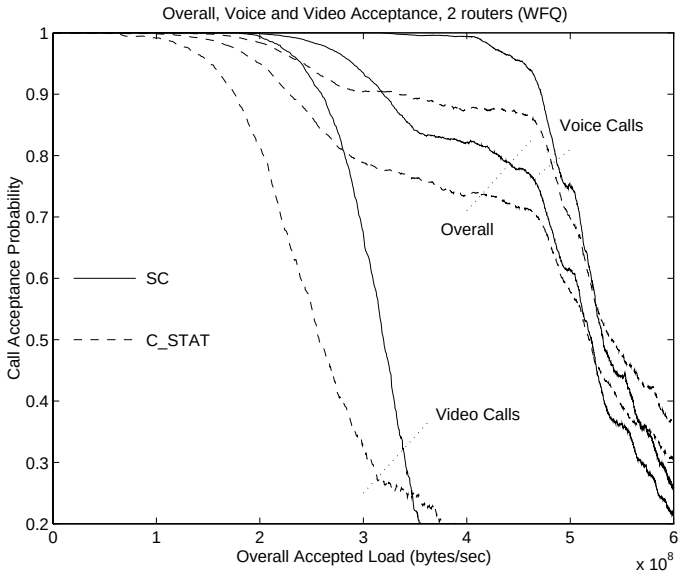

Figure 9: Call acceptance probabilities for heterogeneous traffic (20% video, 80% voice), WFQ server, 155 Mb/s link speeds. 95% confidence intervals for any curve are no greater than 8.9%.

(video acceptance goes down to 0), followed by a leveling off (100% of voice is still being accepted), followed by the second drop (when voice acceptance starts falling).

The crossover of the C_STAT algorithm relative to the others can again be explained in terms of fairness (or lack of it). Routing a video channel over a long path consumes a lot of resources. An algorithm such as C_STAT (which favors channels requiring short paths) permits a much larger number of channels to be accepted at high network loads. However, rejections occur much earlier (particularly for video traffic) for the static algorithm, due to its inability to avoid local congestion.

The WFQ admission control policies accepts a significantly higher load than even the Stop&Go policy under these conditions. The primary reason is that a fixed set of packet transmission rates (frame sizes) was available with the Stop&Go policy. Calls that don't fit one of these rates exactly will cause rate overallocation, which is wasted bandwidth. Our implementation of WFQ placed no constraints on rates and allocated exactly the bandwidth needed by a call.[5]

5.3 Other Experiments

In the experiments described above, the end-to-end delay bound for all calls was assumed to be 350 ms. These delay bounds are easy to meet for all traffic conditions/networks/admission control policy combinations found in those experiments. Another experiment was run for

[5] A simple experiment (not shown) demonstrated this is the reason for the difference in the accepted load. The earlier version of this paper [23] showed little difference between Stop&Go and WFQ for a smaller and slower network.

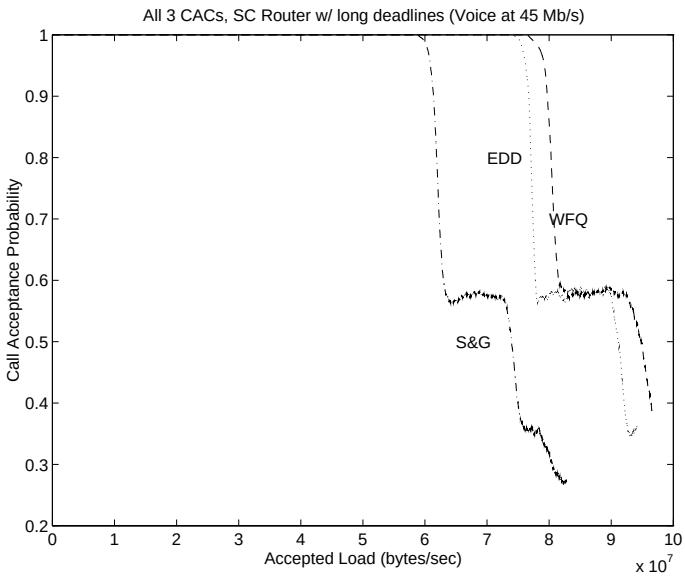

Figure 10: Comparison of admission control policies for loose end-to-end delay bounds, homogeneous traffic (voice only), SC routing algorithm, 45 Mb/s link speeds. 95% confidence intervals for any curve are no greater than 6.7%.

purposes of comparing the admission control algorithms under conditions where the delay bounds are much more stringent. This experiment simulated a network with 45 Mb/s links, voice traffic only, and one routing algorithm (SC), for each of the three admission control policies. The network was simulated once for calls with long delay bounds (350 ms), and once for calls with short delay bounds (varied uniformly between 20 and 50 ms). The results are shown in Figures 10 and 11.

Since this network has a diameter of 7 hops and each hop has a propagation delay of 3 ms, just the propagation time for some paths will be 20 ms or more. The delays due to framing (Stop&Go, equation 4) and fair queuing (WFQ, equation 6) for voice at 45 Mb/s are substantial, and must be added to this. Achieving the shorter delay bounds is therefore rather challenging.

For the given conditions, the behavior of the EDD policy is essentially unaffected by the shorter delay bounds; EDD is considered to be flexible in its ability to cope with stringent delay bounds. The Stop&Go policy, on the other hand, suffers a substantial drop in accepted load. An examination of the delay bound condition for Stop&Go (Equation 4) indicates smaller end-to-end delays can only be achieved by assigning channels to smaller frame sizes. Reserving one packet slot every T_g seconds for a channel in which peak packet interarrival time is greater than T_g naturally implies bandwidth is wasted, leading to the lower accepted loads. WFQ shares with Stop&Go a dependence between the allocated rate and achievable delay bound, as indicated by Equation 6. To achieve lower delays with WFQ also requires overallocating the bandwidth assigned to a channel. Our implementations of WFQ and Stop&Go limited the rate assigned to a channel to be no more than a maximum amount. As a result, channels for which the required delay bound can only be met by allocating more

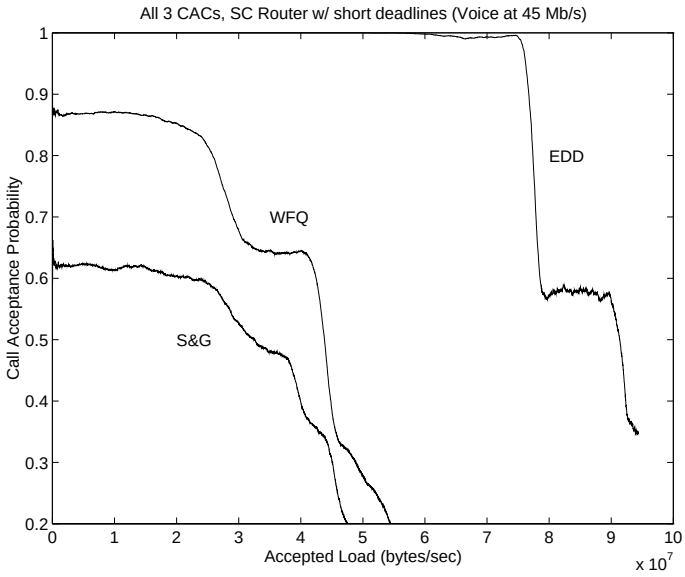

Figure 11: Comparison of admission control policies for stringent end-to-end delay bounds, homogeneous traffic (voice only), SC routing algorithm, 45 Mb/s link speeds. 95% confidence intervals for any curve are no greater than 8.8%.

than the maximum rate are simply rejected. This leads to immediate rejections even for an unloaded network. It also tends to reject channels requiring longer paths (since delay is proportional to hop-count, by Equation 6), which is a form of unfairness.

All of the experiments discussed above were conducted on the network shown in Figure 2. It is possible that network topology strongly influences the behavior of the routers. Accordingly, an experiment was conducted on a network whose topology was a four-dimensional hypercube. A hypercube is a completely symmetric network with a high degree of connectivity between the nodes. The network link speeds were 155 Mb/s and the EDD admission control policy was used. The results are shown in Figure 12. The results can be compared with Figure 4, for identical conditions except the difference in network topology.

The differences in router performance are more substantial for the hypercube network. This is because in a highly-connected network there are more opportunities for optimization, justifying the use of a more intelligent router. The relative ranking of the algorithms (except for C_DYN) remains the same; algorithm SC does particularly well in this network. The steep fall-off in acceptance probability is due to the symmetric nature of the hypercube. In such a network, a load-balancing routing algorithm results in all links filling up at essentially the same moment. C_DYN does particularly well for this network topology. Under these conditions (uniform network bandwidth and topology, loose delay bounds, homogeneous traffic characteristics), a dynamic algorithm based solely on utilization compares very favorably with real-time algorithms.

Section 2 mentioned that a common criticism of deterministic admission control/ multiplexing methods is the low average utilization they can achieve. This is directly due to the high peak/mean ratio of multimedia traffic sources, such as compressed video and voice.

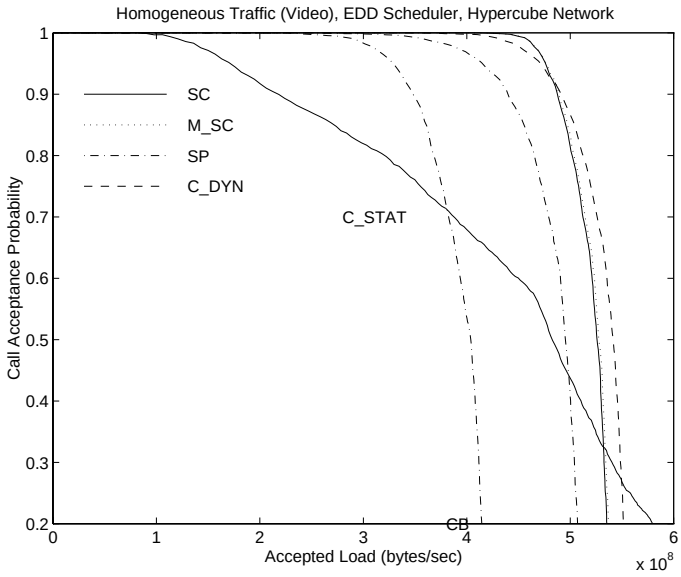

Figure 12: Call acceptance probabilities for a hypercube network topology, with homogeneous traffic (video only), EDD admission control method, and 155 Mb/s link speeds. 95% confidence intervals for any curve are less than 1.5%.

	EDD	**Stop&Go**	**Stop&Go with Shaping**	**WFQ**
Peak Traffic Rates	.637	.676	.690	.687
Average Traffic Rates	.212	.226	.382	.230

Table 3: Average link utilizations for peak and average source traffic rates, for homogeneous traffic (video only), the SC routing algorithm, and 155 Mb/s link speeds. Measured at a call acceptance probability of 75%. 95% confidence intervals for all numbers are less than 1.5%.

Traffic shaping reduces this peak/mean ratio, which should lead to improved utilization even while preserving strong guarantees on quality-of-service. For the voice and video models of this paper, the effective bandwidth computations yield the curves shown in Figure 13.

An experiment was conducted to investigate this hypothesis. This experiment was run for the original network of Figure 2, with 155 Mb/s link speeds, the Stop&Go admission control algorithm, and video traffic only, with traffic shaping of the video sources. A simple calculation indicates that no channel should incur an end-to-end delay in the network of greater than 50 ms for these conditions, leaving 300 ms (out of the specified delay bound of 350 ms) for traffic shaping. From Figure 13, the effective bandwidth for a video source can be decreased by more than 25% for a shaper delay of 300 ms and acceptable loss of 10^{-8}. Figure 14 shows the results of this experiment, and should be compared with Figure 4. The use of shaping substantially improves the absolute performance of all routers, while not affecting their relative performance. Table 3 shows the increase in average network utilization which is achieved for one specific acceptance probability. The use of traffic shaping improves average utilization by almost 70% for this specific combination of network conditions and

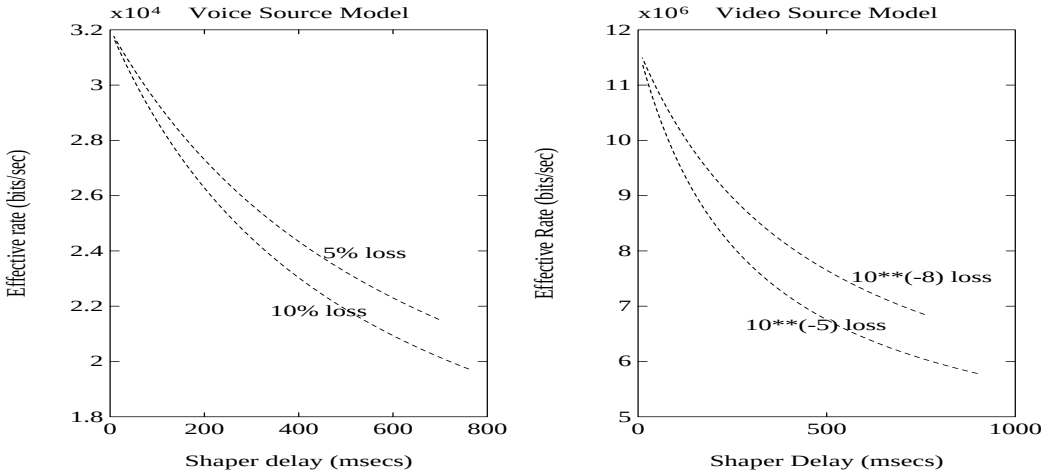

Figure 13: Effective bandwidth vs. shaper-induced delay for voice (peak = 32Kb/s, mean = 11.24Kb/s) and video (peak = 11.7Mb/s, mean = 3.85Mb/s) source models.

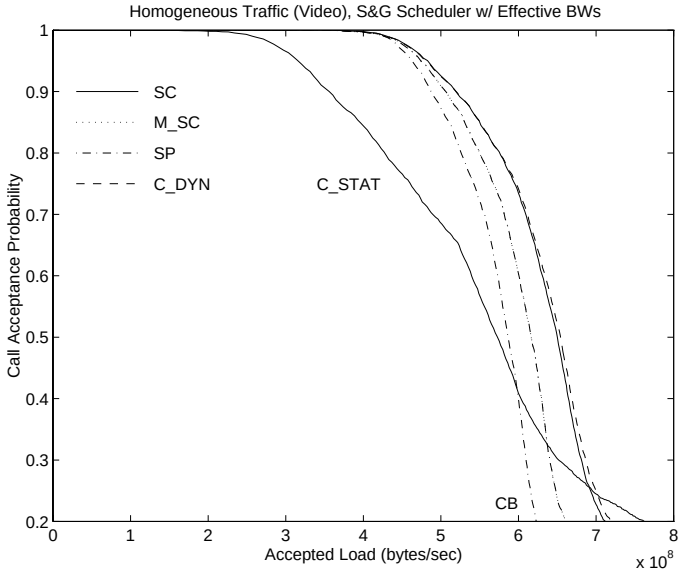

Figure 14: Call acceptance probabilities for homogeneous traffic (video) with traffic shaping, Stop&Go admission control, and 155 Mb/s link speeds. 95% confidence intervals for any curve are less than 2.4%.

admission control and routing policies.

6 Conclusion

We experimentally evaluated the performance of several routing algorithms and three admission control policies for multimedia traffic in packet-switched networks. Our study included all routing algorithms which have been suggested or proposed as being well-suited for this purpose, and a number of new algorithms as well.[6] The new routing algorithms specifically target the real-time requirements of multimedia. The algorithms were compared on their ability to accept new calls at a given network load. This is the first comprehensive study of real-time routing and deterministic admission control algorithms for voice and video traffic. It is also the first study of the close relationship between routing and admission control.

We found that a dynamic real-time routing algorithm (SC) which addresses the constraints of the admission control policy performs best overall. The benefits of this router are greatest when quality-of-service requirements are difficult to meet, the network is highly connected, and admission control constraints are stringent. A conventional dynamic algorithm based solely on utilization also performs reasonably well, although it is not as fair to channels requiring long paths. Sequential (circuit-switched) routing algorithms perform noticeably worse than shortest-path algorithms.

The relative and absolute performance of the admission control policies depended very much on quantization effects, end-to-end delay requirements, and the traffic mix. Careful attention should be given to these factors in selecting and implementing any one of these policies, or low utilizations will result. The use of traffic shaping, if allowable under the user-specified bounds on delay, can significantly improve the link utilizations achieved by deterministic admission control / multiplexing policies.

There are several important open problems raised by this work:

- In our simulations, the routing algorithm always has complete, up-to-date knowledge of the state of the network. In reality, there is some lag between the time a call is accepted, and the time routers at distant nodes become aware of the change in the network this causes. Typically, this requires some bandwidth to propagate information about the network state, and causes some loss in router performance. The impact is greatest when each channel represents a significant fraction of a link's bandwidth. Implementation issues such as this need to be studied.

- The performance of these routing algorithms with statistical admission control and multiplexing policies should be investigated. The time-consuming nature of packet-level simulation, and the difficulty of fixing the packet delays or losses with statistical methods, make this a challenging problem.

- Statistical admission control methods yield high network utilization, while deterministic admission control methods produce strong end-to-end QoS guarantees. A different

[6] A total of 10 different routing algorithms, including 3 not shown in this paper, were evaluated in our experiments.

approach is to combine the best features of statistical and deterministic methods, so that both high network utilization and strong end-to-end guarantees can be achieved. A first attempt at accomplishing this can be found in [24].

- For some experimental results, we had to qualify our findings by making statements about fairness. What is needed is an objective way to quantify fairness, and mechanisms to enforce it, so that routers can be decisively compared without such qualifying statements.

Issues such as these must be resolved to fully realize the potential of packet-switched networks for multimedia applications.

Acknowledgements: We thank D. Agrawal and I. Viniotis of N. C. State University, and the anonymous referees, for their helpful advice and comments.

References

[1] H. Ahmadi, J. Chen and R. Guerin, "Dynamic Routing and Call Control in High-Speed Integrated Networks," *Proc. of the 13th International Teletraffic Congress*, Copenhagen, June 1991, pp. 397-403.

[2] C.M. Aras, J. Kurose, D. S. Reeves, and H. Schulzrinne, "Real-time Communication in Packet-switched Networks," *Proceedings of the IEEE*, Vol.82, No.1, pp. 122-139.

[3] D. Bertsekas and R. Gallager, *Data Networks*, Englewood Cliffs, New Jersey:Prentice-Hall, 2nd Ed 1992.

[4] D.D. Clark, S. Shenker and L. Zhang, "Supporting real-time applications in an integrated services packet network: architecture and mechanism," *Proc. SIGCOMM '92*, August 1992, pp.14-26.

[5] A. Demers, S. Keshav, and S. Shenker, "Analysis and Simulation of a Fair Queuing Algorithm", *Internetworking: Research and Experience*, Vol.1, No.1, September 1990, pp. 3-26.

[6] A.I. Elwalid and D. Mitra, "Effective Bandwidth of General Markovian Traffic Sources and Admission Control of High Speed Networks," *IEEE/ACM Trans. Networking*, Vol.1, No.3, June 1993, pp.329-343.

[7] D. Ferrari and D.C. Verma, "A Scheme for Real-time Channel Establishment in Wide-area Networks," *IEEE Journal on Selected Areas in Communications*, Vol.8, No.3, April 1990, pp.368-379.

[8] M. Garey and D. Johnson, "Computers and Intractability: A Guide to the Theory of NP-Completeness," W. H. Freeman, 1979.

[9] M. Gerla, H. W. Chan, J. Boisson De Marca, "Routing, Flow Control, and Fairness in Computer Networks," *Proc. IEEE Intl. Conf. on Computer Communications*, May 1984, pp.1272-1275.

[10] A. Girard, "Routing and Dimensioning in Circuit-Switched Networks", Addison-Wesley Publ. Co., 1990.

[11] S.J. Golestaani, "A Framing Strategy for Congestion Management," *IEEE Journal on Selected Areas in Communications*, Vol.9, No.7, September 1991, pp.1064-1077.

[12] W. D. Grover, "The SelfHealing Network: A Fast Distributed Restoration Technique for Networks Using Digital Cross-Connect Machines," *Proc. Globecom '87*, IEEE, Dec. 1987, pp.1090-1095.

[13] R. Guerin, H. Ahmadi and M. Naghshineh, "Equivalent Capacity and its Application to Bandwidth Allocation in High-Speed Networks," *IEEE Journal on Selected Areas in Communications*, Vol.9, No.7, September 1991, pp. 968-981.

[14] R-H. Hwang, J.F. Kurose and D. Towsley, "The effect of Processing Delay and QoS Requirements in High Speed Networks," *Proc. IEEE INFOCOM '92*, pp. 160-162.

[15] D.D. Kandlur, K.G. Shin and D. Ferrari, "Real-time Communication in Multi-hop Networks," *Proc. IEEE Intl. Conf. on Distributed Computing*, 1991, pp.300-307.

[16] V. Kompella, J. Pasquale and G. Polyzos, "Multicast Routing for Multimedia Communications," *ACM/ IEEE Transactions on Networking*, vol.1, No.3, June 1993, pp.286-292.

[17] B. Maglaris, D. Anastassiou, P. Sen, G. Karlsson and J. Robbins, "Performance Models of Statistical Multiplexing in Packet Video Communications," *IEEE Trans. Communications*, vol.36, no.7, July 1988, pp.834-844.

[18] K. Murakami, "Near-Optimal Virtual Path Routing for Survivable ATM Networks," *Proc. INFOCOM '94*, IEEE, June 1994, pp.208-215.

[19] A.K. Parekh and R.G. Gallager, "A generalized processor sharing approach to flow control in integrated services networks-the multiple node case," *Proc. INFOCOM '93*. pp.521-530.

[20] C.J. Paris and D. Ferrari, "A Dynamic Connection Management Scheme for Guaranteed Performance Services in Packet-Switching Integrated Services Networks," Tenet Technical Report TR-93-005, Computer Science Division, University of California at Berkeley, 1993.

[21] C. Partridge, *Gigabit Networking*, Addison-Wesley, 1993.

[22] R. Perlman, "Interconnections: Bridges and Routers", Addison-Wesley, 1992.

[23] S. Rampal, D.S. Reeves and D.P. Agrawal, "An Evaluation of Routing and Admission Control Algorithms for Multimedia Traffic in Packet-switched Networks," *Proc. of the 5th IFIP Conf. on High-Performance Networking*, Grenoble, June 1994, pp. 77-92.

[24] S. Rampal and D. S. Reeves, "End-to-End Guaranteed QoS with Statistical Multiplexing for ATM Networks," in *Performance Modelling and Evaluation of ATM Networks*, D. Kouvatsos, ed., Chapman-Hall Publ., 1995. Also published in *Proc. of the 2nd IFIP Workshop on Performance Modelling of ATM Networks*, Bradford England, July 1994.

[25] D. Yates, J.F. Kurose, D. Towsley and M. Hluchyj, "On per-session end-to-end delay distributions and the call-admission problem for real-time applications with QoS requirements," *Technical Report*, Dept. of Computer Science, Univ. of Massachusetts, Amherst, 1993.