On the Use of Quality of Service in IP over ATM

Bruce A. Mah[1]
bmah@CS.Berkeley.EDU
The Tenet Group
Computer Science Division
University of California at Berkeley
Berkeley, CA 94720-1776
(510) 642-8905 (voice)
(510) 642-5775 (fax)

ABSTRACT

The future Internet will very likely include some components based on Asynchronous Transfer Mode (ATM), as well as existing network technologies. This environment produces the possibility of using one of the strengths of ATM (namely, quality-of-service guarantees) to improve the performance of Internet traffic. In this report, we document some of our work in progress on the use of performance-guaranteed ATM virtual circuits to carry IP datagrams. We are currently addressing the implications of various IP-over-ATM strategies on network performance, particularly those aspects relating to quality of service, multiplexing, and virtual circuit management. We are evaluating these performance effects using INSANE, a new object-oriented, discrete-event network simulator.

1 Introduction

The construction and deployment of Asynchronous Transfer Mode (ATM) networks are a recent development in the field of computer communications. Integrating this new technology into the existing Internet requires schemes for managing the transmission of IP datagrams over ATM networks. Such schemes ideally will take advantage of the strengths of ATM, such as its capability for supporting quality of service guarantees, while effectively bridging the gap between the data forwarding models of ATM (virtual circuits) and the Internet (datagrams).

We are investigating the use of performance-guaranteed ATM virtual circuits to carry Internet traffic in a heterogeneous internetwork environment. In particular, we are examining the performance effects of different quality of service mappings, policies for multiplexing IP conversations onto virtual circuits, and virtual circuit management policies. We are currently constructing a network simulator to explore these issues with different simulated workloads.

Section 2 provides some background and motivation for this work. In Section 3, we discuss some related work. We present the issues of our research in Section 4. Finally, we describe our completed and current work in Section 5 and Section 6, respectively.

2 Background and Motivation

Asynchronous Transfer Mode (*ATM*) is a new network technology designed for “integrated services” networks capable of carrying multimedia data as well as conventional computer data traffic [Alles93]. ATM is a connection-oriented service that transfers small, fixed-sized packets called *cells* through a switched-based network. Network protocols called *ATM Adaptation Layers* (*AALs*) fragment larger, variable-sized packets into cells for transmission and reassemble them upon arrival at their destinations. With the appropriate scheduling disciplines in the network

1. Support for this research was provided by (NSF/DARPA) NCR-8919038, the Corporation for National Research Initiatives, (DOE) DE-FG03-92ER-25135, AT&T Bell Laboratories, Digital Equipment Corporation, Hitachi, Ltd., International Computer Science Institute, Mitsubishi Electric Research Laboratories, and Pacific Bell.

switches and support in the signalling software, ATM networks have the potential to provide real-time[2] performance guarantees, such as bounds on bandwidth and packet loss [ATM93]. These performance guarantees are likely to be necessary for many network applications, such as digital audio and video [Ferrari90].

Although a fairly new innovation, ATM is currently gaining popularity. However, it is uncertain whether or not ATM will become a ubiquitous, dominant network technology. The existing installed base of Local Area Networks (LANs) such as Ethernet is considerable; replacing these networks with ATM will be costly, and in some cases unnecessary. For the foreseeable future, it appears that large-scale connectivity will continue to involve multiple, heterogeneous networks, and appropriate internetwork layer protocols. In many circumstances, ATM networks will be used as backbones, connecting existing LANs as described by [Cavanaugh92].

In the current Internet, the solution to forwarding data through such a heterogeneous internetwork is provided by the Internet Protocol (IP) [Postel81a]. IP is almost entirely independent of the subnet technology used; it makes few assumptions about the nature of individual subnets. IP packets can traverse many different types of subnets (including ATM networks) without either the senders or receivers being aware of the details of the networks encountered along the path. Unlike ATM, IP is a datagram protocol and does not require the establishment of connections before data can be sent. Although IP does have some provisions to support a weak form of quality of service (QOS) [Almquist92], such facilities are only rarely supported in practice [Bohn93] [Stevens94].

As ATM and the Internet will likely co-exist in the near future, it is desirable that hosts using these two types of networks be able to exchange data. One approach to interoperability is to use an ATM network (with an appropriate adaptation layer) as a datalink layer, similar to Ethernet and FDDI, as shown in Figure 1. This service is commonly referred to as *IP-over-ATM*.

Transport Layer	Transmission Control Protocol (TCP)		User Datagram Protocol (UDP)
Internetwork Layer	Internet Protocol (IP)		
Datalink Layer	Ethernet	FDDI	AAL / ATM

FIGURE 1. The Internet Protocol Suite and Datalink Layers.

An interesting aspect of this problem concerns the use of quality of service guarantees in an ATM network to the benefit of *IP conversations*[3]. The issue of ATM QOS also impacts multiplexing and virtual circuit management. In particular, different multiplexing policies trade off protection of the performance given to individual IP conversations with the possibility of better use of resources reserved to a given virtual circuit. Different virtual circuit management strategies trade off longer-lived resource reservations for delays caused by virtual circuit setup and teardown.

3 Related Work

Much of the work to date in the field of IP over ATM has concerned various paradigms for the service, and how they affect the issues of addressing and routing. Studies dealing with multiplexing and virtual circuit management in IP over ATM have been performed in the context of best-effort virtual circuits, but have not addressed quality of service issues within ATM networks. Various solutions for supporting quality of service or performance guarantees in an

2. By “real-time”, we refer to a service that can provide mathematically provable bounds or guarantees on performance.

3. We informally define an IP conversation as a set of packets that have some logical connection with each other. For example, the set of packets belonging to a single TCP/IP connection can be thought of as a single conversation.

internetwork have been proposed, but have not dealt with the specific characteristics of ATM subnets and how they are used to transport IP traffic.

3.1 Paradigms for IP over ATM

Several different paradigms for the use of ATM in the Internet environment have been defined. [Laubach94], which proposes a “classical model of IP over ATM”, is typical of the *subnet models*, in which an ATM network or internetwork is used as a subnet in an IP internetwork. In the classical model, ATM-attached hosts establish virtual circuits between themselves to carry IP datagrams, either on demand or permanently at startup time. An address resolution server with a well-known address handles the translation between IP addresses and ATM addresses. The ATM network is seen as an opaque “routing cloud”. Routers at the edge of the ATM network (or elsewhere on the Internet) are oblivious of its internal details and topology. Figure 2 illustrates the subnet model. A description and discussion of this and other IP over ATM models proposed to the Internet Engineering Task Force can be found in [Cole95].

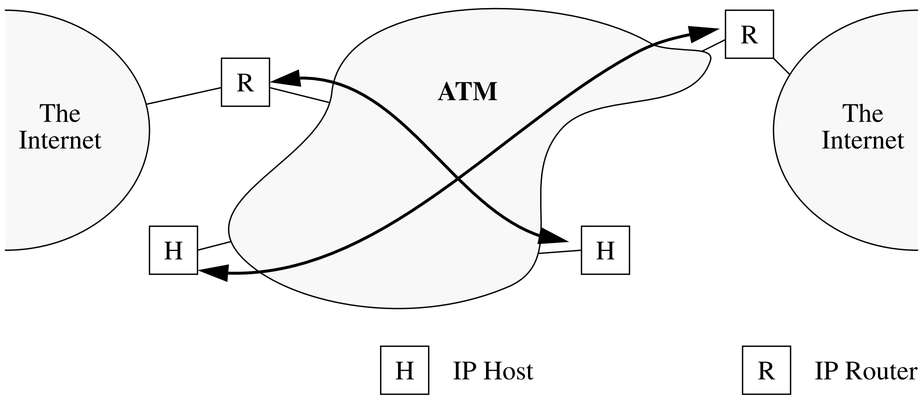

FIGURE 2. Subnet Model of IP over ATM. ATM-attached hosts and routers establish virtual circuits between each other, but the ATM network appears as an opaque “routing cloud”.

Another model, *LAN Emulation* (LANE or LE), has been proposed by the ATM Forum [ATM95]. LANE makes an ATM network appear as a traditional shared-medium LAN such as Ethernet or FDDI. Because this emulation is transparent to the network layer, LANE supports many different network protocols, in addition to IP. For the purposes of this research, there are few differences between an ATM network running LANE and one running IP over ATM.

A third model, discussed in [Box93] and [Omundsen94], proposes *connectionless servers* within ATM networks to handle datagrams such as those generated by IP. In this model, end hosts establish virtual circuits to connectionless servers within the ATM network, which forward datagrams in much the same way as IP routers. Virtual circuits between connectionless servers carry datagrams towards their destinations. This approach has the advantage of multiplexing virtual circuits among many different network conversations, but, due to this sharing, has difficulties providing performance guarantees or protection between competing traffic sources. For this reason, we exclude this model from our analysis, but present it here for completeness. This model is shown in Figure 3.

3.2 Traffic Multiplexing Policies

Several studies have examined the issue of multiplexing datagrams for different IP conversations over the same ATM virtual circuit; this research, however, has been performed in the context of best-effort virtual circuits (with no associated quality of service parameters). [Cáceres92] explores several different policies for multiplexing TCP [Postel81b] conversations in ATM networks. The environment under investigation was a wide-area ATM network using best-effort service to support conventional data traffic carried by TCP/IP. This work was based on traces of WAN traffic from three different Internet sites. The simulations performed in this study suggest that the best multiplexing policy is to establish a virtual circuit per traffic type between each pair of hosts; however this policy was only considered in the context of best-effort virtual circuits.

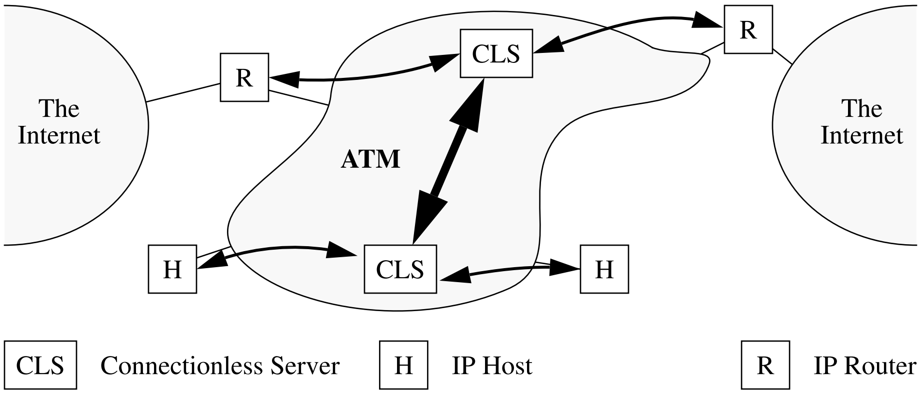

FIGURE 3. Connectionless Server Model of IP over ATM. Hosts and routers establish virtual circuits to "connectionless servers", which forward packets between themselves on multiplexed virtual circuits.

Another study based on actual Internet traffic measurements, [Claffy94], found that many wide-area conversations are short. It recommends that such conversations be routed through a mesh of *Permanent Virtual Circuits* (PVCs) in order to avoid the latency incurred by ATM virtual circuit establishment. The study recommends that on-demand *Switched Virtual Circuits* (SVCs) are only necessary for conversations with different priorities or QOS requirements, or for conversations whose high resource utilization would adversely impact the performance of other traffic over the PVC mesh (e.g. high bitrate video).

Commercially available ATM LANs, such as the FORE Systems ATM LAN described in [Biagioni93], typically multiplex all communication between a given pair of hosts on a single virtual circuit (either a PVC or SVC). This approach has the advantage of implementation simplicity, but has the disadvantage of giving identical treatment to all packets between a given host pair.

3.3 Virtual Circuit Management

The time to establish a virtual circuit may be relatively long, especially in wide-area backbones with long propagation times. Therefore it may be useful to keep virtual circuits in existence even when they are not actively being used to transmit data. A simple solution that eliminates virtual circuit setup time altogether is to create a mesh of PVCs between all pairs of endpoints. This approach is recommended by both [Cáceres92] and [Claffy94] but has inherent scaling problems in the case of large ATM subnets.

When SVCs are being established and torn down dynamically, it may be possible to amortize the connection setup time by caching virtual circuits in the hope they can be reused for other IP conversations. The utility of such caching is dependent on the arrival and duration of IP packets and conversations, as well as the characteristics of those conversations. [Lund95] describes an adaptive strategy for computing virtual circuit holding times, which involves gathering an empirical distribution of packet interarrivals. The approach is fairly simple to implement and has shown promising results in trace-driven simulations.

The IP-over-ATM implementations in current production ATM LANs typically use either SVCs with static timeouts or PVCs. The FORE System ATM LAN ties the lifetime of virtual circuits to a host's ARP cache, which results in a timeout of fifteen minutes. Thus, any virtual circuit which is idle for longer than fifteen minutes is torn down [Biagioni93].

Within the Internet community, [Perez95] documents the ATM signalling necessary to support the IP-over-ATM service of [Laubach94].

3.4 Quality of Service in an Internetwork

Although no prior work has been published on the use of QOS in ATM subnets supporting IP, the problem of providing QOS in an internetworking environment such as the Internet has received a considerable amount of attention. For the most part, existing IP networks do not provide any quality of service support. All packets and conversations are treated identically. However, some work has been done with the Type of Service (TOS) bits in the IP header [Almquist92] or the precedence field [Bohn93] to express the priority assigned to a datagram. The next version of IP (variously referred to as *IPv6* or *IP: Next Generation*) contains support for a *Flow ID*, which can be used to identify datagrams as belonging to a particular flow and thus eligible to receive a particular treatment by routers [Bradner95].

The Integrated Services (IS) model for the Internet includes a model known as *predictive service* [Braden94]. Networks based on predictive service use the current state of traffic in the network when performing admission control tests. Predictive service offers a weak quality of service model, as the admissions tests are based on current network conditions, not on worst-case assumptions (applications are assumed to adapt to changing network conditions). The resource reservation protocol RSVP is designed to support predictive service in the Internet [Zhang95].

Various solutions exist to address quality of service considerations in non-IP internetworks. For example, networks based on algorithms and protocols such as the Tenet Real-Time Protocol Suite [Banerjea95] offer mathematically provable end-to-end real-time performance guarantees. They require the applications to specify their requirements to the network in advance. Admission control tests are used to limit the number and type of real-time connections allowed into the network, and provide deterministic or statistical performance guarantees. These guarantees hold even under “worst-case” conditions.

4 Research Issues

We consider an ATM network being used as a subnet that is part of a larger, heterogeneous IP internetwork. We assume that Internet routers and gateways will forward data to and from the ATM network and other attached, external networks. They perform all normal IP routing functions, forwarding IP datagrams through ATM virtual circuits (either PVCs or SVCs or both) as required. ATM-attached routers will use virtual circuits directly established among themselves to carry IP packets.

We presume that the internetwork will carry integrated services traffic. In addition to conventional data traffic (such as file transfers and remote logins), future integrated networks will be carrying continuous media digital audio and video data, similar to that being transmitted across the MBONE, a virtual network to support IP multicast in the Internet [Macedonia94]. Each of these different data types will have different characteristics and will thus require different services from any networks attempting to furnish an appropriate quality of service [Cáceres91] [Mah94b].

We consider several interesting issues related to the use of QOS-guaranteed virtual circuits to carry IP datagrams under these circumstances. These issues relate to the QOS requirements to be assigned to virtual circuits, multiplexing policies, and policies for virtual circuit management.

4.1 Quality of Service Requirements

The first problem concerns the mapping of a set of IP datagrams onto an ATM virtual circuit with the appropriate QOS. The correct quality of service to be used for a conversation will depend on the application. For example, interactive applications such as `telnet` and `rlogin` require low delays to be useful. Bulk file transfers (such as those performed by `ftp`) work best over high throughput connections. In many cases the QOS will be implied, based on pre-existing knowledge about applications. Currently, Internet applications are not required to specify their QOS requirements (indeed, there is no currently widely-accepted standard for doing so, though RSVP [Zhang95] is a popular contender). There are, however, several methods for a network to determine the QOS requirements of a stream of IP datagrams:

- By examining the TCP or UDP port numbers or other higher-layer information, a host or router at the edge of the ATM network may be able to determine the application, and hence the appropriate quality of service. This approach relies on many applications in the Internet using well-known ports and the network usage of common applications being well-known.

- By monitoring the throughput for a given conversation over time, the network may be able to compute an appropriate set of requirements for an ATM virtual circuit (for example peak and average throughput requirements). Such an adaptive scheme is of course only useful when an IP conversation lasts long enough to permit such measurements.

- The application may be able to send some sort of QOS request indicating its requirements. Such a message could, for example, be contained within an IP option of a data packet or be sent using a signalling protocol such as RSVP [Zhang95] or the Real-Time Channel Administration Protocol (RCAP) [Mah93].

- A default set of parameters (perhaps "best effort with no resource reservation") is necessary for the case in which no QOS can be determined for a given conversation.

We recognize that this facility will not provide end-to-end performance guarantees to IP conversations (at least not in the case of a heterogeneous internetwork). However we believe that this approach can improve network performance for applications traversing an ATM backbone network, where resources are presumably more scarce than in a local area environment.

4.2 Multiplexing with QOS Considerations

Another issue, already explored to some detail for best-effort networks, is how a given virtual circuit should be multiplexed between streams of arriving datagrams. Intuitively, multiplexing a virtual circuit among many IP conversations should yield better utilization of that virtual circuit's reserved network resources due to statistical multiplexing, but at the expense of a decreased level of protection between conversations sharing the same ATM virtual circuit.

For example, we can examine the characteristics of actual audio-video conferences on the Internet, as described in [Mah94b], and consider their transmission across an ATM subnet[4]. In these conferences, usually only one user was speaking at a time. A single ATM virtual circuit could be multiplexed among all of the involved UDP/IP conversations and used to carry the audio for the entire conference. Because only one audio source was active at once, that virtual circuit would only require enough resources to support one sender at a time. By contrast, most users were seen to be sending video data continuously throughout; in order to protect the performance of each of the video streams, each of the associated UDP/IP conversations would need to be assigned its own ATM virtual circuit.

4.3 Virtual Circuit Management with QOS Considerations

The use of performance-guaranteed virtual circuits, which reserve resources within an ATM network, raises some additional problems with respect to the creation and use of virtual circuits for sending IP traffic. For instance, the approach of creating a PVC mesh becomes less attractive in an ATM subnet attempting to provide QOS support to IP traffic. A fixed set of PVCs cannot truly be expected to provide the quality of service suitable for a possibly unknown traffic load. Moreover, a fixed mesh of QOS-guaranteed virtual circuits ties up resources unnecessarily, as the connections are not *a priori* known to be necessary.

It is therefore important and interesting to consider the dynamic creation and teardown of performance-guaranteed SVCs for use by IP. An important tradeoff, made more so by the implications of resource reservations, concerns the lifetime of ATM virtual circuits. To reduce the effects of virtual circuit setup time on latency, it is generally desirable to cache unused SVCs in the hope that they will be needed again (possibly for a different IP conversation). However, to keep the real-time utilization of the network high, it is important to free up the resources reserved by SVCs as soon as they are no longer needed. Any virtual circuit management policy must attempt to balance these two goals, possibly incorporating the characterizations of individual types of IP conversations as well.

Multiplexing a single virtual circuit among multiple IP conversations may introduce some additional issues and have some new implications for the fixed timeout strategies suggested in [Cáceres92] and [Claffy94] or the adaptive strat-

---

4. The conferences studied in [Mah94b] were sent using IP multicast. The support of IP multicast over an ATM network introduces additional issues and is beyond the scope of this study. The discussion in this example merely addresses the question of how to forward the data from an audio/video conference across an ATM subnet.

egies proposed by [Lund95]. For example, a virtual circuit's lifetime may depend on the arrival patterns of traffic for multiple IP conversations or on the relationships between different IP conversations multiplexed over that virtual circuit.

5 Completed Work

We are currently in the process of evaluating the relative performance of various IP-over-ATM policies addressing the issues discussed in Section 4. We have completed two pieces of work towards this end. First, we have built and tested a set of modifications to the IP service of an existing ATM WAN to support various multiplexing and virtual circuit management policies. Second, we have taken measurements of multimedia applications on the Internet, which can be used to generate one component of a workload for our evaluation.

5.1 XUNET II Implementation

We have built and tested enhancements to the IP service on XUNET II to support different IP-over-ATM policies. XUNET II is a testbed designed to support research on high-speed wide-area networks [Fraser92]. It is an experimental network which runs ATM over DS3 (45 Mbps) links to connect FDDI networks at eight sites across the continental United States. Each site has an IP router based on a Silicon Graphics Iris 4D/310S workstation to forward packets between the attached local networks and the long-distance DS3 links. The physical topology of the XUNET II ATM links is shown in Figure 4.

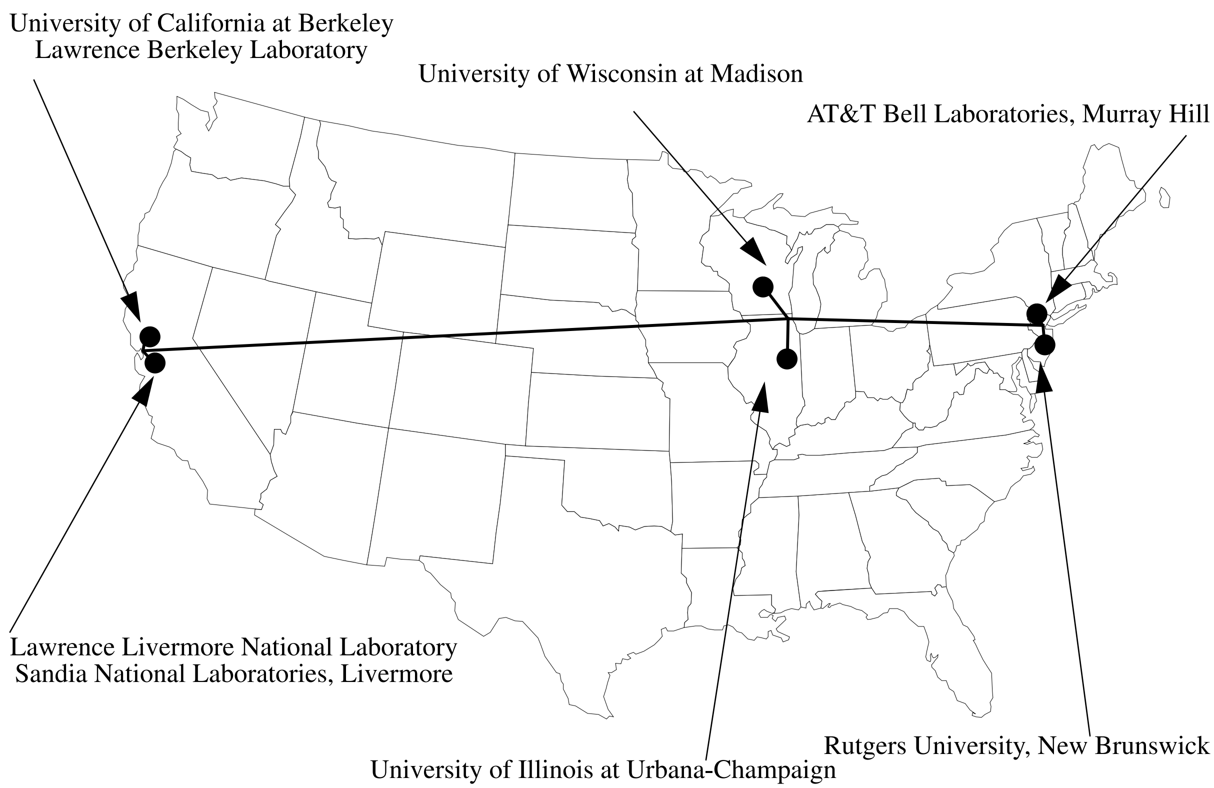

FIGURE 4. Physical Topology of XUNET II backbone links.

Currently, the production version of the XUNET II software sends IP datagrams over a PVC mesh, with a single virtual circuit between every pair of routers. We have modified the drivers in the XUNET routers to enable them to discriminate between packets belong to different IP conversations. We distinguish different IP conversations by a *conversation key*, which consists of the source and destination IP address, IP type-of-service field, the transport layer protocol, and (where applicable) the source and destination port numbers at the transport layer. The applicable header fields of a TCP/IP packet are shown in Figure 5 (analogous fields are used for the datagram protocol UDP [Postel80] and other protocols layered on top of IP).

<table>
<tr><td>Version</td><td>Hdr Len</td><td>Preced</td><td>TOS</td><td colspan="4">Total Length</td></tr>
<tr><td colspan="4">ID</td><td>Flags</td><td colspan="3">Fragment Offset</td></tr>
<tr><td colspan="2">TTL</td><td colspan="2">Protocol</td><td colspan="4">IP Header Checksum</td></tr>
<tr><td colspan="8">Source Address</td></tr>
<tr><td colspan="8">Destination Address</td></tr>
<tr><td colspan="4">Source Port</td><td colspan="4">Destination Port</td></tr>
<tr><td colspan="8">TCP Sequence Number</td></tr>
<tr><td colspan="8">TCP Acknowledgment Number</td></tr>
<tr><td>Hdr Len</td><td>Reserved</td><td colspan="2">Flags</td><td colspan="4">Window Size</td></tr>
<tr><td colspan="4">TCP Checksum</td><td colspan="4">Urgent Pointer</td></tr>
</table>

FIGURE 5. TCP/IP Header Fields Used for Conversation Keys. The top header is for IP, while the bottom header is for TCP. The white fields above are used to identify individual IP conversations.

This key allows us to map all the packets from, for example, a single TCP/IP connection or a single UDP/IP conversation onto their own virtual circuit. By controlling a function used to determine the equivalence of conversation keys, we can vary the IP over ATM multiplexing policy used in the XUNET routers. For example, we can ignore the protocol field and port numbers to multiplex all packets between a pair of hosts onto a single virtual circuit.

We have also given the routers the ability to set up ATM switched virtual circuits on demand, in response to arriving IP datagrams. Although our switched virtual circuits currently time out after a statically-configured interval, we can easily support a more sophisticated virtual circuit management policy.

Preliminary measurements and experimentation show that the new, enhanced XUNET IP driver works correctly and that the added processing needed to perform packet classification does not significantly degrade the performance of the XUNET router. Throughput measurements of the original and enhanced XUNET driver are shown in Figure 6. As can be seen, the performance of the enhanced driver is roughly the same as the original (in some cases, the throughput has been improved due to some unrelated tuning of the driver). Figure 7 shows round-trip measurements between two XUNET routers at Bell Labs[5]. More details can be found in [Mah94a].

5.2 MBONE Measurements

Conventional data applications on the Internet have been extensively measured and studied. However, there exists little information on the network usage of newer multimedia applications, such as `nv` and `vic`. To gain some data and insight into their network traffic patterns, we have taken traffic measurements of multicast, multimedia traffic sent over the MBONE, a virtual network to support Internet-wide multicasting [Mah94b]. We have collected traces of long audio/video conferences (as well as other MBONE sessions), which will help us to determine appropriate parameters for ATM virtual circuits carrying these types of traffic.

6 Work in Progress

In the course of implementing the enhanced IP driver briefly described in Section 5.1, we realized that we needed a more controlled and flexible environment that XUNET II can currently provide. Therefore we have decided to com-

5. These figures do not include the time required to establish SVCs. In fact, XUNET II has an abnormally long virtual circuit setup time compared to other ATM networks; establishment of a new SVC takes approximately 250–300 ms per hop. By contrast, we measured a SVC setup time of about 70 ms for a path across a single Synoptics (now Bay Networks) ATM LAN.

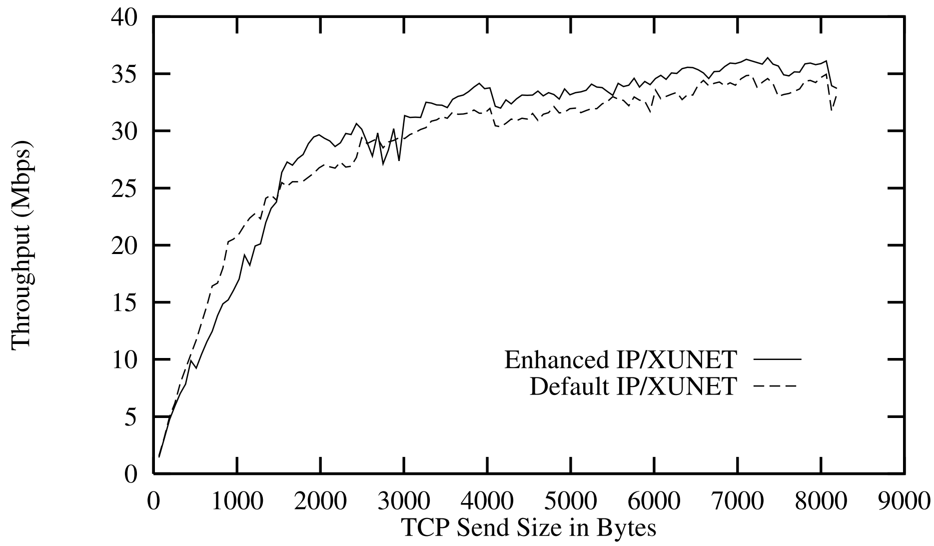

FIGURE 6. Throughput of Default and Enhanced XUNET IP Drivers for Varying Sized TCP Sends.

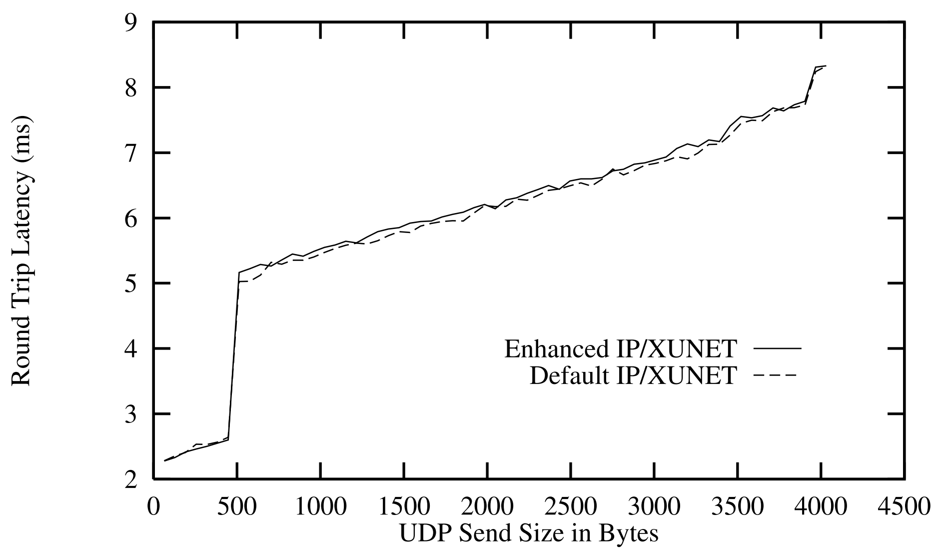

FIGURE 7. Round-Trip Latency of Default and Enhanced XUNET IP Drivers for Varying Sized UDP Sends.

plete our evaluation of the performance effects of various IP-over-ATM alternatives via simulation. This section provides some details on the state of our current work.

6.1 Environment

We are simulating a wide-area ATM backbone network connecting local-area (i.e. campus-wide) LANs or internetworks. A simple proposed network topology is shown in Figure 8, in which LANs at four sites are connected by wide-area ATM links. This topology was chosen to be similar to that of XUNET II.

The ATM links in this proposed topology have a bandwidth of 1.5 Mbps (T1), an attempt to reach a compromise between the desire to simulate high-speed networks and the desire to reduce the computational complexity of the simulation. Propagation delays on these links are on the order of tens of milliseconds, consistent with wide-area networks crossing the continental United States[6].

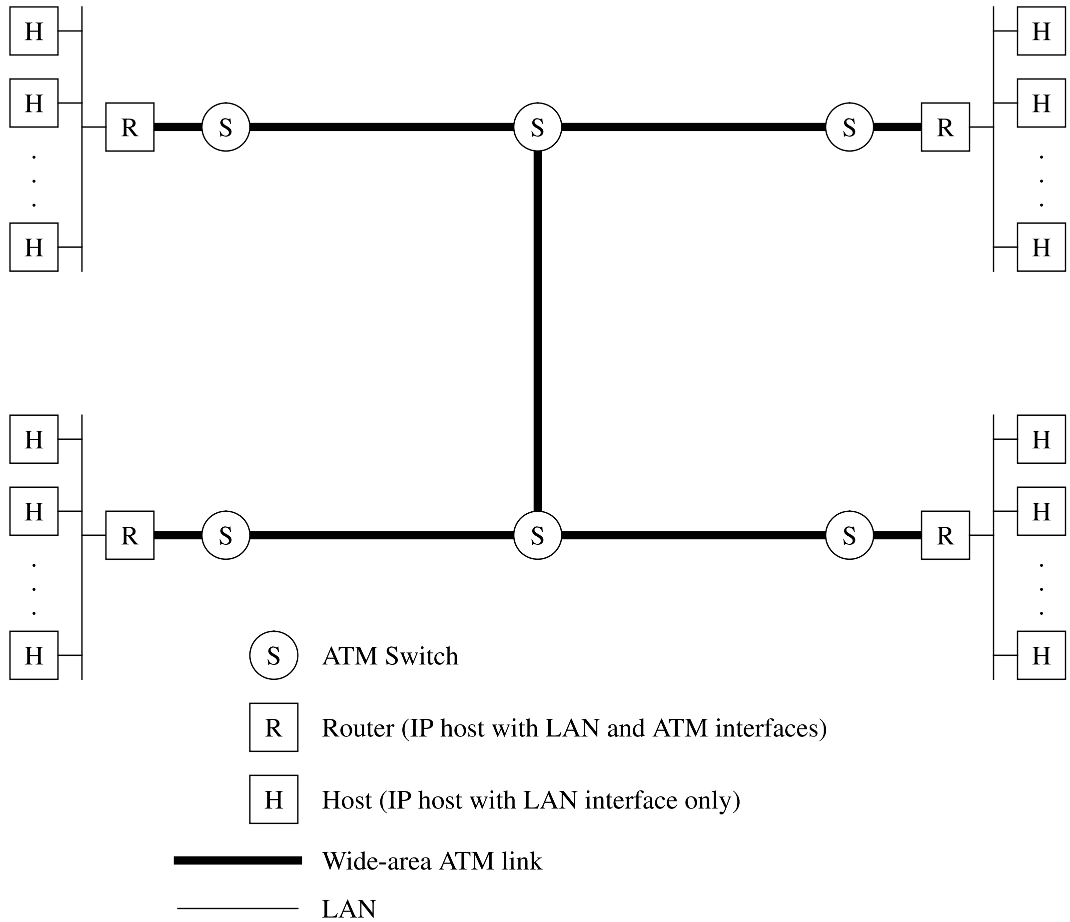

FIGURE 8. Proposed topology of experiments.

The local area networks are simulated broadcast-style networks of arbitrarily high bandwidth. Since we assume that bandwidth and other network resources will be relatively plentiful in local-area (as opposed to wide-area) environments, we feel the LANs should not present any bottlenecks to end-to-end network performance. Traffic sources and sinks are on the hosts on the LANs. There will be a small number of servers (the destinations of connections) and a larger number of clients (initiators of connections) on each local-area subnet, with a few tens of hosts per LAN.

6.2 Workload

To drive the simulation, we will simulate the traffic generated by several popular WAN applications and higher-layer protocols. These include:

- `ftp`: The File Transfer Protocol. Traditionally, this application has been representative of bulk data transfer programs.

- `telnet`/`rlogin`: These two remote login protocols are the two most common of their type seen on the Internet. They have traditionally been used as examples of typical interactive applications.

---

6. The round-trip time between the XUNET routers in Berkeley, CA and Murray Hill, NJ is approximately 55 ms [Pendarakis95].

- `http`: The Hyper-Text Transfer Protocol, used by clients and servers in the World Wide Web.

- `nv/vic`: These two applications are used to transmit low-bitrate digital video across the MBONE, a virtual network used to provide IP multicast services to large portions of the Internet.

We will use models based on traffic traces to simulate the behavior of each individual conversation, using an approach similar to that taken by `tcplib` [Cáceres92]. While we may be able to use `tcplib`, verbatim, for `telnet/rlogin` conversations and some attributes of `ftp` conversations. additional data will be required to simulate `http` and `nv/vic` conversations. (The traffic traces collected in [Mah94b] include data on `nv` and `vic` conversations from which a conversation workload may be constructed.)

The issue of how to generate an aggregate workload from individual conversations is slightly more problematic. Data in [Cáceres91] shows that the traffic generated by different Internet sites differs significantly, and that selecting a "typical" site workload is difficult. In addition, [Paxson95] documents that the Poisson processes usually used to model connection arrivals are not always a good match for actual traffic. Finally, [Paxson94] reports that the characteristics of a site's Internet traffic can vary widely over time, especially with the introduction of a new popular application, such as the World Wide Web. For the purposes of this study, it will probably be necessary to select some "reasonable" workload (or a small number of workloads); further study is required to determine this model exactly.

6.3 ATM Policies

Section 4 described design alternatives for the various IP-over-ATM policies. We plan to investigate a subset of these policies, each defined by the cross-product of several subpolicies. We will examine two different quality of service subpolicies:

- *No QOS*: All virtual circuits are "best effort".

- *QOS*: Each application type (i.e. `telnet/rlogin`) is mapped onto a set of performance parameters appropriate to that application's traffic and performance requirements. There are, of course, many possible mappings; we will initially pick one, possibly to be refined and varied at a later time.

We will consider three different multiplexing subpolicies:

- *Virtual circuit per router pair*

- *Virtual circuit per conversation*

- *Virtual circuit per application type* (i.e. one virtual circuit for all FTPs passing through a pair of routers)

Finally, we examine three different virtual circuit management subpolicies:

- *PVC*: A mesh of permanent virtual circuits is established to carry IP packets.

- *SVC*: Switched virtual circuits, with some timeout policy to be determined, are used to carry IP packets.

- *SVC/cache*: Similar to SVC, but with the additional feature that virtual circuits can be cached and reused to carry the packets carried by other IP conversations.

The three types of subpolicies can be viewed as three parameter axes, which are not entirely orthogonal. First we observe that for an IP-over-ATM service using PVCs, a discussion of QOS and multiplexing policies is pointless. No QOS can be set for an unknown workload traversing a fixed set of virtual circuits. Moreover, for a PVC mesh, the sheer number of virtual circuits required will probably force a multiplexing policy of one virtual circuit per router pair.

Second, we observe that when performing VC per router pair multiplexing, it is impossible to assign a meaningful QOS to such a virtual circuit because the traffic between the routers is unknown.[7]

The values of the policy axes, subject to the above constraints, results in the "policy space" shown in Figure 9.

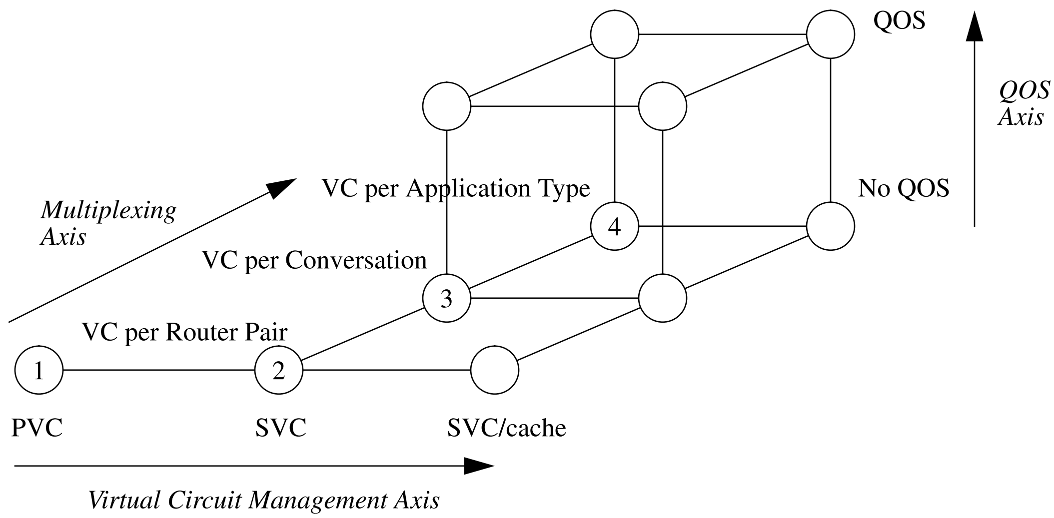

FIGURE 9. IP-over-ATM Policy Space.

Certain points in this policy space are of particular interest:

- The PVC policy (1) represents the simplest possible IP-over-ATM policy. It is used by XUNET II in its default configuration. Other ATM networks (such as BAGNET) use this policy if interoperability problems make providing an SVC or SVC/cache service infeasible.

- The SVC/per router pair/no QOS policy (2) is used by several commercial ATM LANs, including the FORE System ATM LAN described in [Biagioni93].

- Our prior enhancements to the XUNET IP service cover the various SVC/no QOS policies (2, 3, 4).

6.4 Attributes of the ATM Network

The ATM Forum has defined a number of traffic classes and associated performance parameters for the support of different qualities of service [ATM93]. Unfortunately, to the best of our knowledge, a set of specific algorithms for supporting these services have not yet been defined. Our simulated ATM network, while not strictly conforming to the ATM Forum standards, uses algorithms whose properties and guarantees are well-known. We rely only on the following attributes of ATM networks:

- ATM is a switch-based network, with point-to-point links between switches.

---

7. This statement is not entirely accurate. We can assign some QOS to the aggregate traffic flow, between the two routers, but the effects on an individual IP conversation are unknown because the traffic is, in general, unknown. Since one of the issues of this research is to investigate the performance aspects with respect to individual IP conversations, this particular situation is not of interest to us.

- ATM is connection-oriented (as opposed to datagram-based). There is some signalling protocol capable of setting up virtual circuits and (if necessary) performing the necessary admission control tests. Our simulated ATM network uses a protocol based heavily on the Real-Time Channel Administration Protocol (RCAP), the signalling protocol used in the Tenet Real-Time Protocol Suite [Mah93].

- ATM can support performance guarantees. Although we make no assumptions for the mechanism for providing guarantees, we do assume a well-defined interface. We are using RCSP [Zhang93] as a scheduling mechanism (along with appropriate admission control tests) to provide performance guarantees over our simulated ATM networks.[8] In the case that no performance guarantees are to be provided, FIFO queueing is used within the network.

6.5 Evaluation

We will measure the performance of the various applications of interest, under different IP-over-ATM policies. Different metrics will be of interest to different applications:

- For `ftp`, throughput on the data connection is likely the primary concern. However, for interactive file transfers, delay may also be a criteria of interest.

- `telnet` and `rlogin` sessions have delay as their primary performance metric. As remote login applications generally do not send large amounts of data, throughput is not an issue.

- `http` clients will be primarily interested in response time, although throughput (in the case of large inlined images, for example) will also be an important criteria. The primary difference between HTTP and FTP is that the transfer of a single World Wide Web page (which can be thought of as a single HTTP *session*) may consist of multiple TCP connections.

- `nv`, `vic`, and other similar video applications require sufficient bandwidth for their video streams. For adaptive applications, the amount of delay jitter incurred in the network may also be a useful metric, as it determines the amount of buffering that the applications may need to provide.

6.6 An Internet Simulated ATM Networking Environment

To perform our evaluation, we have constructed an *Internet Simulated ATM Networking Environment* (*INSANE*, for short). INSANE is an object-oriented, discrete-event network simulator. The simulation core and a set of basic, atomic objects are all written in C++ [Stroustrup91]. These atomic objects include simple classes such as queues, as well as other building blocks (such as IP, TCP, and UDP protocol modules).

The various atomic objects all export Tcl [Ousterhout94] commands to provide for the creation and setup of new objects. These commands may be invoked from within a configuration file (in fact, INSANE's configuration file is merely a Tcl program). The use of Tcl as a configuration language allows a high degree of flexibility in constructing simulations. Higher-level, composite objects (such as ATM switches) can be created through the use of Tcl procedure, which create and configure atomic objects (such as input and output switch ports and other switch building blocks). Because the Tcl scripts are all external to the simulator itself, different simulation configurations can be rapidly constructed and tested.

7 Summary

Recent trends in networks (in particular, higher bandwidths and switch-based networks) have led to several new networking technologies suitable for integrated services networks which are candidates for both local area and wide area networks. Asynchronous Transfer Mode (ATM) is one such technology that is gaining acceptance in both industry

8. In the specific case of XUNET II, the scheduling discipline proposed in [Saran93] can be used to provide performance guarantees.

and research fields. One issue of particular importance is how to use ATM networks as a portion of the Internet, a heterogeneous internetwork containing a mix of existing network technologies. The differences between ATM networks (virtual circuit-based, with the possibility of performance guarantees) and the Internet (datagrams, with best-effort service) create some interesting, new challenges for research.

We are investigating several issues related to the use of quality of service (QOS) guaranteed virtual circuits in ATM networks to carry Internet traffic. Specifically, we are examining the effects and performance of different quality of service mapping strategies, various multiplexing policies, and some strategies for virtual circuit management. We are currently preparing to evaluate the performance of these various policies using INSANE, a new event-driven network simulator.

8 References

[Alles93] A. Alles. *ATM in Private Networking, A Tutorial*, Hughes LAN Systems, Mountain View, CA, 1993.

[Almquist92] P. Almquist. “Type of Service in the Internet Protocol Suite”, *Internet Request for Comments 1349*, July 1992.

[ATM93] ATM Forum. *ATM User-Network Interface Specification*, Version 3.0, PTR Prentice Hall, Englewood Cliffs, NJ, September 1993.

[ATM95] ATM Forum. *LAN Emulation Over ATM*, Version 1.0, January 1995.

[Banerjea95] A. Banerjea, D. Ferrari, B. Mah, M. Moran, D. Verma, and H. Zhang. “The Tenet Real-Time Protocol Suite: Design, Implementation, and Experiences”, to appear, *IEEE/ACM Transactions on Networking, Volume 3*, Number 6, December 1995.

[Biagioni93] E. Biagioni, E. Cooper, and R. Sansom, “Designing a Practical ATM LAN”, *IEEE Network*, March 1993.

[Bohn93] R. Bohn, H. Braun, K. Claffy, and S. Wolff. “Mitigating the Coming Internet Crunch: Multiple Service Levels via Precedence”, to appear, *Journal of High Speed Networks*, 1994.

[Box93] D. Box, D. Hong, and T. Suda. “Architecture and Design of Connectionless Data Service for a Public ATM Network”, *Proceedings of INFOCOM ‘93*, San Francisco, CA, March 1993.

[Braden94] R. Braden, D. Clark, and S. Shenker. “Integrated Services in the Internet Architecture: An Overview”, *Internet Request for Comments 1633*, June 1994.

[Bradner95] S. Bradner and A. Mankin. “The Recommendation for the IP Next Generation Protocol”, *Internet Request for Comments 1752*, January 1995.

[Cáceres91] R. Cáceres, P. Danzig, Sugih Jamin, and D. Mitzel. “Characteristics of Wide-Area TCP/IP Conversations”, *Proceedings of SIGCOMM ’91*, Zürich, Switzerland, September 1991.

[Cáceres92] R. Cáceres. “Multiplexing Traffic at the Entrance to Wide-Area Networks”, PhD Dissertation, University of California, Berkeley, CA, December 1992.

[Cavanaugh92] J. Cavanaugh and T. Salo. “Internetworking with ATM WANs”, Minnesota Supercomputer Center Technical Report, December 1992.

[Claffy94] K. Claffy. “Internet Traffic Characterization”, PhD Dissertation, University of California at San Diego, 1994.

[Cole95] R. Cole, “IP over ATM: A Framework Document”, *Internet Draft draft-ietf-atm-framework-doc-04.ps*, July 1995.

[Ferrari90] D. Ferrari. “Client Requirements for Real-Time Communications Services”, *Proceedings of the International Conference on Information Technology*, October 1990, Tokyo, Japan; also *IEEE Communications Magazine*, Volume 28, Number 11, pp. 65-72, November 1990; also *Internet Request for Comments 1193*, November 1990.

[Fraser92] A. G. Fraser, C. Kalmanek, A. Kaplan, W. Marshall, and R. Restrick. “Xunet 2: A Nationwide Testbed in High-Speed Networking”, *Proceedings of IEEE INFOCOM ’92*, Firenze, Italy, May 1992.

[Laubach94] M. Laubach. “Classical IP and ARP over ATM”, *Internet Request for Comments 1577*, January 1994.

[Lund95] C. Lund, S. Phillips, and N. Reingold. “Adaptive Holding Policies for IP over ATM Networks”, *Proceedings of IEEE INFOCOM ’95*, Boston, MA, April 1995.

[Macedonia94] M. Macedonia and D. Brutzman. “Mbone Provides Audio and Video Across the Internet”, *IEEE Computer*, April 1994.

[Mah93] B. Mah. “A Mechanism for the Administration of Real-Time Channels”, MS Report, University of California, Berkeley, CA, March 1993.

[Mah94a] B. Mah. “Enhancements to the XUNET IP Service”, unpublished technical memorandum, AT&T Bell Laboratories, Murray Hill, NJ, July 1994.

[Mah94b] B. Mah. “Measurements and Observations of IP Multicast Traffic”, Technical Report UCB/CSD-94-858, University of California, Berkeley, CA, December 1994.

[Omundsen94] D. Omundsen, A. R. Kaye, and S. Mahmoud. “A Pipelined, Multiprocessor Architecture for a Connectionless Server for Broadband ISDN”, *IEEE/ACM Transactions on Networking*, Volume 2, Number 2, April 1994.

[Ousterhout94] J. Ousterhout. *Tcl and the Tk Toolkit*, Addison-Wesley, Reading, MA, 1994.

[Paxson94] V. Paxson. “Growth Trends in Wide-Area TCP Connections”, *IEEE Network*, July/August 1994.

[Paxson95] V. Paxson and S. Floyd. “Wide Area Traffic: The Failure of Poisson Modeling”, *IEEE/ACM Transactions on Networking, Volume 3*, Number 3, June 1995.

[Pendarakis95] D. Pendarakis. Personal communication, July 1995.

[Perez95] M. Perez, F. Liaw, A. Mankin, E. Hoffman, D. Grossman, and A. Malis. “ATM Signalling Support for IP over ATM”, *Internet Request for Comments 1755*, February 1995

[Postel80] J. Postel, ed. “User Datagram Protocol”, *Internet Request for Comments 768*, August 1980.

[Postel81a] J. Postel, ed. “Internet Protocol”, *Internet Request for Comments 791*, September 1981.

[Postel81b] J. Postel, ed. “Transmission Control Protocol”, *Internet Request for Comments 792*, September 1981.

[Saran93] H. Saran, S. Keshav, C. Kalmanek, and S. Morgan. “A Scheduling Discipline and Admission Control Policy for Xunet 2”, *Proceedings of the Fourth International Workshop on Network and Operating System Support for Digital Audio and Video*, Lancaster, England, November 1993.

[Stevens94] W. R. Stevens. *TCP/IP Illustrated, Volume 1: The Protocols*, Addison-Wesley, Reading, MA, 1994.

[Stroustrup91] B. Stroustrup. *The C++ Programming Language*, Addison Wesley, Reading, MA, 1991.

[Zhang93] H. Zhang and D. Ferrari. “Rate-Controlled Static-Priority Queueing”, *Proceedings of IEEE INFOCOM ‘93*, San Francisco, CA April 1993.

[Zhang95] L. Zhang, R. Braden, D. Estrin, S. Herzog, and S. Jamin. “Resource ReSerVation Protocol (RSVP)—Version 1 Functional Specification”, *Internet Draft draft-ietf-rsvp-spec-07.ps*, July 1995.