Performance Experiences in a Wide Area ATM Network *

Luiz A. DaSilva, Joseph B. Evans, Douglas Niehaus, Victor S. Frost, Roelof Jonkman, Beng Ong Lee, Georgios Y. Lazarou

Information and Telecommunication Technology Center
Department of Electrical Engineering & Computer Science
The University of Kansas
Lawrence, KS 66045

Abstract

Experiments in Asynchronous Transfer Mode (ATM) wide area network (WAN) environments are discussed in this work. The lessons learned through practical experiences with ATM systems and how these can be applied to improving performance are presented. Factors ranging from the transport protocol to the equipment used and the network conditions are considered.

1: Introduction

With the recent deployment of the first wide area ATM networks (ATM WANs), experiments in this environment have become possible. This paper discusses lessons learned through practical experiences with ATM and how these can be applied to improving performance.

Throughput that is one or two orders of magnitude below link speed has sometimes been observed on newly deployed ATM systems. The problem is not that ATM networks perform below expectations, but that many network parameters must be appropriately set in order to obtain the best possible performance. Some of the most important and well-known [13] factors, which will be further discussed in the following sections, relate to protocols, hosts and networks.

Most ATM connections currently utilize TCP/IP as the transport protocol. When using TCP/IP over high speed WANs, throughput may be limited by factors such as excessive retransmissions due to cell losses at the edge of the network, inefficiency due to Maximum Transmission Unit (MTU) size, and TCP window management [2]. In most cases, these limiting factors can be avoided with careful tuning and occasional modifications to application code.

Hosts are often the limiting factor when transmitting or receiving over high-speed networks. Characteristics to be considered include CPU and memory speed, operating system, system parameters and load.

Finally, such factors as propagation delay, switch buffer size, link speeds, bandwidth mismatches, Usage Parameter Control (UPC) and congestion avoidance methods will determine the maximum throughput that can be expected.

Many of these factors have already been discussed in detail in the authors' previously published work [5, 6, 7, 14]; this paper describes recent experiments that address some of the remaining factors (a tutorial including some of these results is to be published in [4]). Experiments described in this paper were conducted over the ACTS ATM Internetwork (AAI). This network provides wide area ATM connectivity between several DoD High Performance Computing centers and the MAGIC and ATDnet gigabit testbeds, with ATM service provided by Sprint [1].

In Sections 2 through 4 of this paper, we describe and discuss our results in each of the three main areas listed above; section 5 summarizes our findings.

2: Protocol Considerations

Most studies published about ATM performance utilize TCP/IP [1, 3, 6, 7, 10, 11, 14], as do most applications currently run over ATM. When TCP is used over high-speed wide area networks, performance is heavily affected by cell losses, MTU size, and window management.

TCP/IP performance is strongly influenced by cell loss rate. The Maximum Transmission Unit (MTU) for IP is approximately 192 cells; therefore, losing a single cell has the effect of requiring up to 191 cells to be retransmitted, in some instances aggravating congestion conditions in the network [7]. The number of packets that need to be retransmitted can be minimized using selective retransmission. Some second generation ATM

*This research is partially supported by DARPA under prime contract DABT63-94-C-0068 to Sprint Corporation.

[1] For more details on the AAI, including a complete list of sites, please refer to http://www.arl.mil/AAI/.

0-7803-3925-8/97 $10.00

switches implement early packet discard, which attenuates the impact of losses on congestion.

Even though small packets minimize the number of cells to be retransmitted in case of packet discard, they limit throughput due to excessive overhead and increased packet processing. The impact of this additional processing is greater at the receiving host, responsible for reassembly and checksum verification. The impact of MTU size on maximum achievable throughput has been widely investigated (see, for instance, [3]).

TCP's window-based flow control tends to limit performance when TCP is used over wide area ATM networks. In order to ensure continuous transmission, the window size must support the bandwidth-delay product for the connection.

3: Host Considerations

Host factors include:

- CPU speed;
- Backplane (I/O bus);
- Memory bandwidth;
- Disk access speed;
- ATM interface and drivers - buffering capabilities will determine whether losses occur at the ATM interface, memory copies, etc.;
- Operating system - the structure of different operating systems vary in their support of TCP/IP window sizes, protocol stacks, MTU sizes, etc.;
- Processor load.

The combination of these factors determine the maximum throughput. A performance benchmark ranking various hosts and operating systems as to their networking capabilities would be very useful; none is available at the moment, partly due to the large number of tunable parameters, whose effects are often non-linear.

Table 1 lists the maximum throughput obtained from several architectures over OC-3 links. These results are affected by host configuration and should not be used for comparison among specific vendor products.

4: Network Considerations

Clearly, the network conditions play an important role on performance. Network factors are addressed through a suite of experiments conducted on an ATM WAN.

To characterize performance in an ATM WAN environment, one needs performance evaluation tools capable of conducting large scale tests that are representative of the traffic generated by current and emerging applications. The experiments described below utilize *NetSpec* [9]. *NetSpec* provides a block structured language for specifying experimental parameters and support for controlling performance experiments containing a large number of connections across a local or wide area network, and is specifically designed to maximize the reproduceability of the experiments.

4.1: Aggregate Throughput Performance

In defining metrics to help consumers compare various ATM equipment, the ATM Forum is considering throughput in terms of lossless, peak and full-load behavior [8]. It is useful, then, to obtain the relationship between throughput and offered load, as shown in Figure 1.

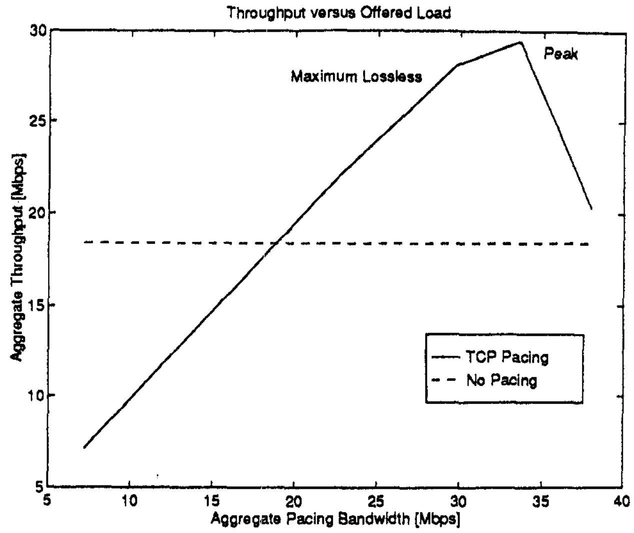

Figure 1. Throughput versus offered load for three simultaneous connections.

The results were generated from an experiment where traffic was simultaneously generated between three pairs of AAI sites. *NetSpec* enabled the variation of offered load by employing application-level pacing to generate constant rate traffic [9]. The three source-destination pairs competed for a common DS-3 link. Therefore, as we increase the offered load, a point is reached where switch buffers overflow and cell losses occur; at this point, a sharp drop in throughput can be seen due to the increased number of retransmissions.

The results refer to the sum of the throughput obtained for each of the three connections. From Figure 1, we can establish the maximum lossless throughput (the maximum throughput obtained without cell losses shows up in the figure as the point where the plot can no longer be approximated by a single straight line, around 28 Mb/s) and the peak throughput (around 29.5 Mb/s). The full-load throughput is obtained by transmitting full rate traffic [2] ; in this case, it was measured as 18.4 Mb/s.

4.2: Impact of Traffic Congestion

To illustrate the effect of congestion, an experiment involving five source-destination pairs, as depicted in Figure 2, was conducted. The values next to each of the five transmitting hosts represent the throughput obtained when each connection was set up individually (top) versus the throughput when traffic was produced simultaneously (bottom), measured in Mb/s. The round-trip time for each connection is provided next to the clocks, and the arrows indicate the direction of the traffic. Congestion had an impact on every connection, and the aggregate throughput decreased 25%, from 73.84 to 55.19 Mb/s. If we increase the offered load for each connection by setting a higher pacing rate, an even sharper drop in throughput occurs, due to cell losses, emphasizing the need for congestion control.

[2]Full rate refers to the transfer of packets from local memory to memory on a remote host as fast as the operating system, interfaces, and network allow.

Architecture	Operating System	Transmit Throughput [Mb/s]	Receive Throughput [Mb/s]
DEC 5000/240	ULTRIX 4.3	54.5	35.8
Pentium-100	Linux 1.3.91	125.0	70.0
SPARC-10/51	SunOS 4.1.3	62.0	99.0
SPARC-20/40	Solaris 2.4	83.0	95.0
SPARC-20/2x125	Solaris 2.4	130.5	115.0
DEC Alpha 3000/400	OSF/1 v3.0	130.6	134.1
DEC Alpha 3000/400	Digital Unix T4.0	133.2	133.5
DEC Alpha 3000/600	OSF/1 v3.2	133.9	133.5

Table 1. Maximum throughput observed for hosts used as receiver or transmitter - measurements were conducted over OC-3 connections. These results are affected by host configuration and should not be used as a benchmark for specific vendor products.

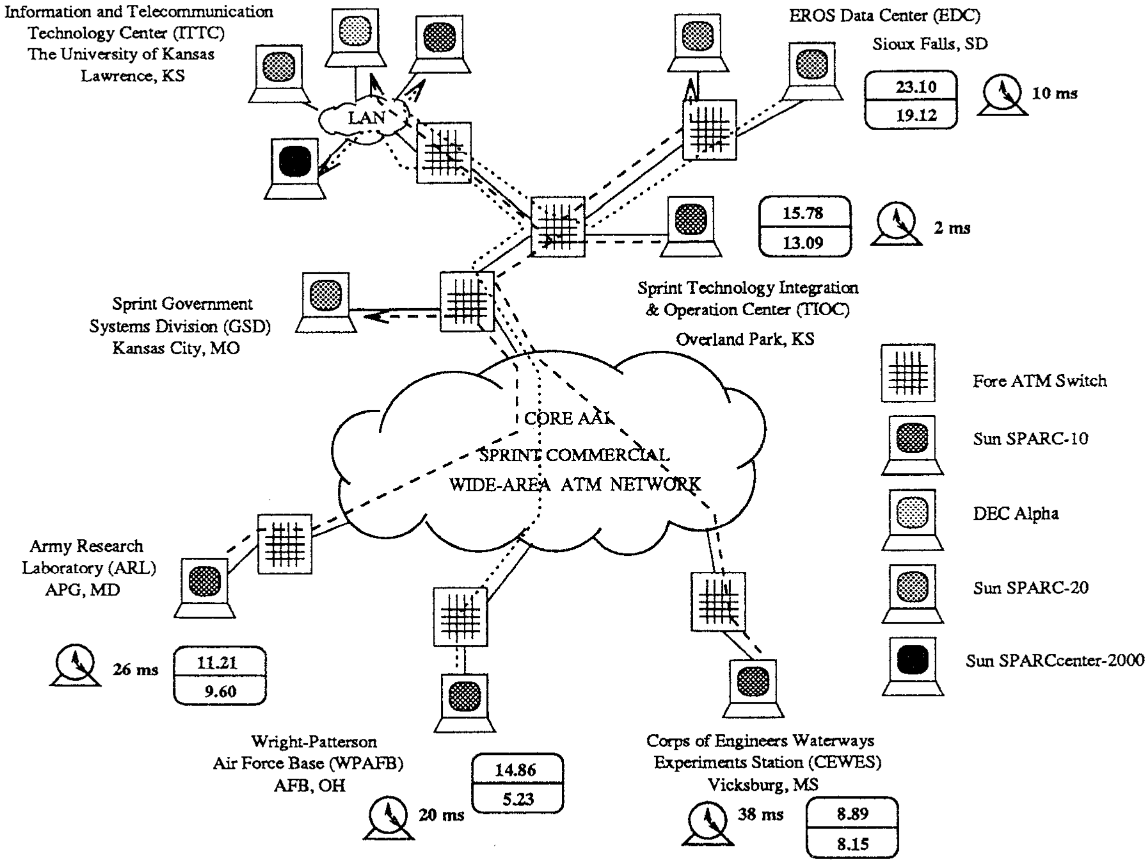

Figure 2. Effect of competing traffic on throughput in a ten-host experiment.

TCP's window-based flow control is not sufficient to avoid congestion of this type, as confirmed by [14].

4.3: Impact of Switch Buffer Resources

To illustrate the impact of switch buffer size, an experiment between ITTC and NRL, as shown in Figure 3, was conducted. Rate mismatches between OC-3 and DS-3 links exist in both directions.

Using *NetSpec*, constant rate traffic was generated at 35 Mb/s (this takes care of the rate mismatch), varying the size of the burst of data transmitted. The relationship between the burst size and throughput is shown in Figure 4.

The asymmetry of the results indicates that the switches used at either end have different buffering capabilities and allows us to roughly estimate these capabilities as 3.5 MB (GSD switch) and 35 KB (NRL switch); these correspond to the knees of the curves in Figure 4. Early packet discard can be used to mitigate this type of problem.

4.4: Impact of Traffic Policing

Traffic policing, or Usage Parameter Control (UPC), has the objective of determining whether ATM cells entering a switch conform to pre-established traffic contracts. Non-conforming cells are either tagged or dropped, ensuring that connections do not exceed their bandwidth reservation.

A set of experiments was performed in order to experimentally evaluate the behavior of TCP/IP over

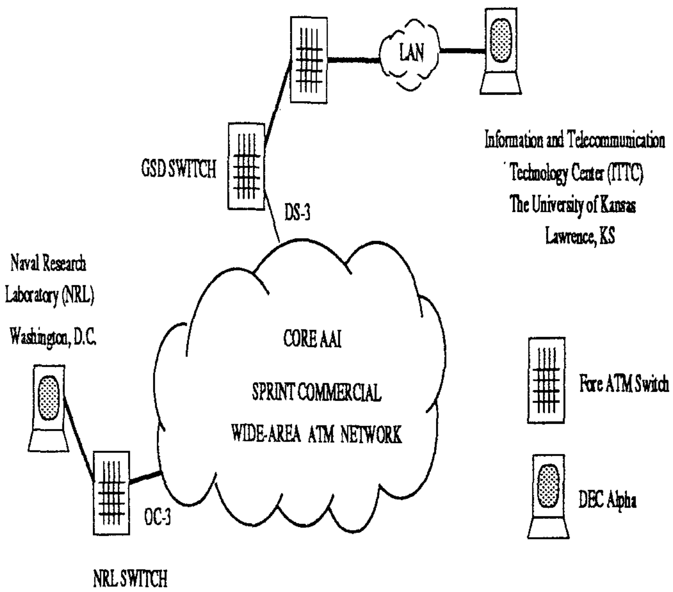

Figure 3. Set up for experiment to estimate switch buffer size.

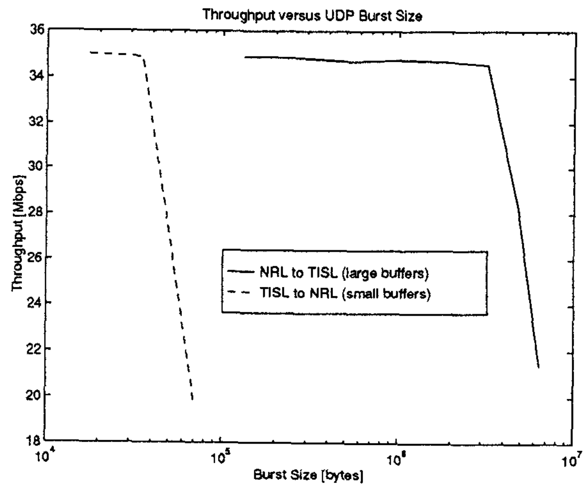

Figure 4. Throughput versus UDP burst size.

ATM with traffic policing. The switch used here employs a combination of the Generic Cell Rate Algorithm (GCRA) [12] and user-configurable parameters in the ATM Management Interface (AMI) to perform the traffic policing function.

The experiments showed that when a source is able to transmit at a higher traffic rate than the one specified in the traffic contract, cell-level pacing is necessary to avoid significant degradation in throughput. TCP rate control mechanism alone, or even combined with application-level pacing, is inadequate to ensure conformance. In the absence of cell-level pacing, contracts with very high Peak Cell Rate (PCR) and/or Cell Delay Variation Tolerance (CDVT) values are needed to ensure the availability of resources for connections with high burstiness. Hosts that are able to transmit faster generally produce burstier traffic, therefore requiring a higher PCR and/or CDVT; this is illustrated in Figures 5 and 6, where the knee of the curve occurs later for the faster host.

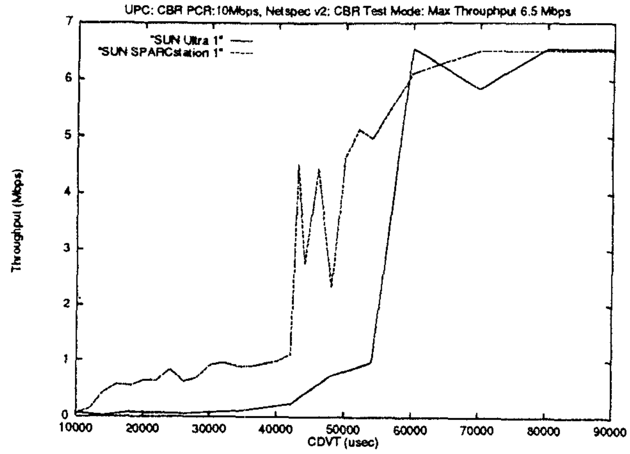

Figure 5. TCP Performance vs CDVT for CBR Contract with PCR=10Mbps

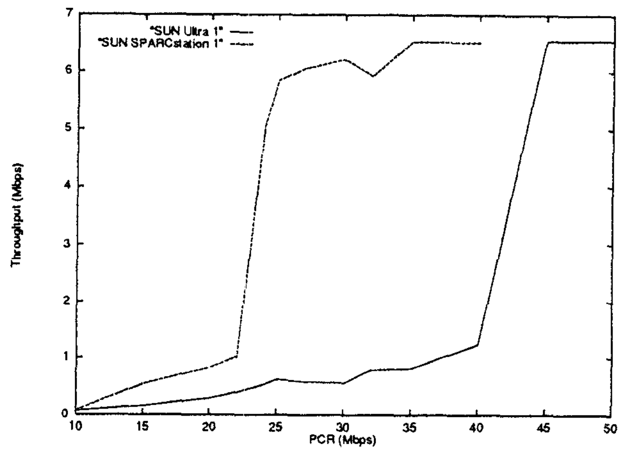

Figure 6. TCP Performance vs PCR with CDVT set to 10000 μsec.

5: Conclusions

This paper discussed the impact of protocol, host, and network factors on ATM WAN performance. In Table 2, we provide a check list for factors that affect performance in such an environment, as well as some useful UNIX commands and utilities that can be used to check and/or modify the appropriate parameters. While the list is certainly not comprehensive, we believe that by following these steps one is able to avoid some of the common factors that can result in poor performance.

Future areas of research include more extensive studies on ABR service, the impact of signalling and the characterization of real-life application performance (video, voice, multimedia, etc.).

A Acknowledgements

The authors would like to acknowledge the active contributions of Chuck Thacker and Hal Murray at Digital Equipment Corporation for the experiments reported in this paper. Sprint's Broadband Operations Center (BBOC) also provided valuable assistance monitoring many of these tests.

The following trademarks are used in this article: *Digital UNIX*, *DEC Alpha 3000* and *DEC AN2* are trademarks of Digital Equipment Corporation; *SunOS*, *Solaris*, *Ultra 1*, *Sun SPARCstation* and *Sun SPARC-*

	Check List	Pointers to Tools
1	What is the round-trip time for the connections being used?	ping
2	If using TCP/IP, is the window (socket buffer size) at least as large as the bandwidth-delay product for the link?	ttcp, netconfig, application code
3	Can the MTU size be increased?	ifconfig
4	Is there significant load on the hosts used for transmission and reception?	top, ps
5	What is the maximum throughput obtained with host/operating system?	ttcp, Netspec
6	If your application accesses the hard disk, what is the maximum access speed?	disktest
7	Are there rate mismatches in the path used? If so, are you pacing the traffic adequately?	atmarp, atmconfig, Netspec
8	Is there any competition for the link(s) or switch(es) used?	SNMP
9	Which route is being followed?	route
10	Are there any cell or packet losses? If so, what is the cause?	netstat, atmstat, SNMP
Notes: ping, ifconfig, route, netstat: standard Unix tools ttcp, Netspec: performance measurement tools atmarp, atmconfig, atmstat: ATM driver-specific tools available for FORE ATM drivers netconfig, top, disktest: custom tools for Unix available in the public domain		

Table 2. Check list for debugging/performance tuning in ATM networks (available from http://www.tisl.ukans.edu/Projects/AAI).

center are trademarks of Sun Microsystems.

References

[1] Olivier Bonaventure, André Danthine, Espen Klovning, and Olivier Danthine. TCP/IP and the European ATM Pilot. In *Proceedings of 1995 Conference on Network Protocols*, September 1995.

[2] John David Cavanaugh and Timothy J. Salo. Internetworking with ATM LANs. Available as http://www.msci.magic.net/docs/magic/ip-atm.ps.

[3] Dan Daly. Results of TCP/IP Traffic Measurements over the ACTS ATM/SONET Channel and Directions for Future Investigation. In *1996 DARPA Workshop on Wide-Area ATM Performance*, Lawrence, KS, June 1996.

[4] Luiz A. Dasilva et al. ATM WAN Performance Tools, Experiments and Results. *IEEE Communications Magazine*, 1997. To be published.

[5] Luiz A. DaSilva, Rick Lett, and Victor S. Frost. Performance Considerations in File Transfers using FTP over Wide-Area Networks. In *Proceedings of the 22nd International Conference for the Resource Management & Performance Evaluation of Enterprise Computing Systems (CMG96)*, San Diego, CA, December 1996.

[6] Joseph B. Evans et al. A 622 Mb/s LAN/WAN Gateway and Experiences with Wide Area ATM Networking. *IEEE Network*, 10(3):40–48, May/June 1996.

[7] Benjamin J. Ewy, Joseph B. Evans, Victor S. Frost, and Gary J. Minden. TCP/ATM Experiences in the MAGIC Testbed. In *Proceedings of the 4th IEEE Symposium on High Performance Distributed Computing*, pages 87–93, August 1995. Available as http://www.magic.net/tcp/overview.html.

[8] Raj Jain. Frame-level Performance Benchmarking Work at the ATM Forum. In *1996 DARPA Workshop on Wide-Area ATM Performance*, Lawrence, KS, June 1996.

[9] Roelof Jonkman. Netspec: A Network Performance Evaluation and Experimentation Tool. Available as http://www.tisl.ukans.edu/ netspec/.

[10] Georgios Y. Lazarou, Victor S. Frost, Joseph B. Evans, and Douglas Niehaus. Using Measurements to Validate Simulation Models of TCP/IP over High Speed ATM Wide Area Networks. In *Proceedings of the 1996 IEEE International Conference on Communications (ICC'96)*, Dallas,TX, June 1996.

[11] Allyn Romanow. TCP over ATM: Some Performance Results. *IEEE Journal on Selected Areas in Communications*, 13(4), May 1995.

[12] Willian Stallings. *ISDN and Broadband ISDN with Frame Relay and ATM.* Prentice Hall, third edition, 1995.

[13] Andrew S. Tanenbaum. *Computer Networks.* Prentice Hall, third edition, 1996.

[14] Hongbo Zhu, Luiz A. DaSilva, Joseph B. Evans, and Victor S. Frost. Performance Evaluation of Congestion Control Mechanisms in ATM Networks. In *Proceedings of the 21st International Conference for the Resource Management & Performance Evaluation of Enterprise Computing Systems (CMG95)*, Nashville, TN, December 1995.