Experiences with TCP/IP over an ATM OC12 WAN

Rebecca L. Nitzan
Brian L. Tierney

Lawrence Berkeley National Laboratory

Abstract

This paper discusses the performance testing experiences of a 622.08 Mbps OC12 link. The link will be used for large bulk data transfer, and as such, of interest are both the ATM level throughput rates and end-to-end TCP/IP throughput rates. Tests were done to evaluate the ATM switches, the IP routers, the end hosts, as well as the underlying ATM service provided by the carrier. A low level of cell loss, (resulting in <.01 % packet loss), decreased the TCP throughput rate considerably when one TCP flow was trying to use the entire OC12 bandwidth. Identifying and correcting cell loss in the network proved to be extremely difficult. TCP Selective Acknowledgement (SACK) improved performance dramatically, and the maximum throughput rate increased from 300 Mbps to 400 Mbps. The effects of TCP slow start on performance at OC12 rates are also examined, and found to be insignificant for very large file transfers (e.g., for a 10 GB file). Finally, a history of TCP performance over high-speed networks is presented.

1.0 Introduction

This paper describes performance testing of an OC12 link between Lawrence Berkeley National Laboratory (LBNL), in the San Francisco Bay Area, and Argonne National Laboratory (ANL), near Chicago. This testing started in the fall of 1998.

There were several goals; one was to determine the capabilities of the OC12 link, which has a theoretical line rate of 622 Mbps, and the equipment in support of this link, such as IP routers and ATM switches. One of the main purposes of this link is to provide support for very large bulk data transfers between LBNL and ANL, so another goal was to evaluate end-to-end TCP/IP protocol performance. Therefore we examined TCP related performance issues, and the effects of various TCP options such as SACK and slow start. In this paper we also explore the history of TCP over large bandwidth-delay product networks, sometimes known as “big fat pipes”.

1.1 Test Configuration

A testbed was constructed that was largely dedicated to these tests (see Figure 1). It was composed of routers, ATM switches, hosts, and an analyzer. In some cases, the ATM switches were shared with other production traffic, however the OC12 ports of the switches were dedicated to the testing.

The OC12 service, supplied by Sprint, was shared with other Sprint customers at a non-specific, yet estimated, rate.

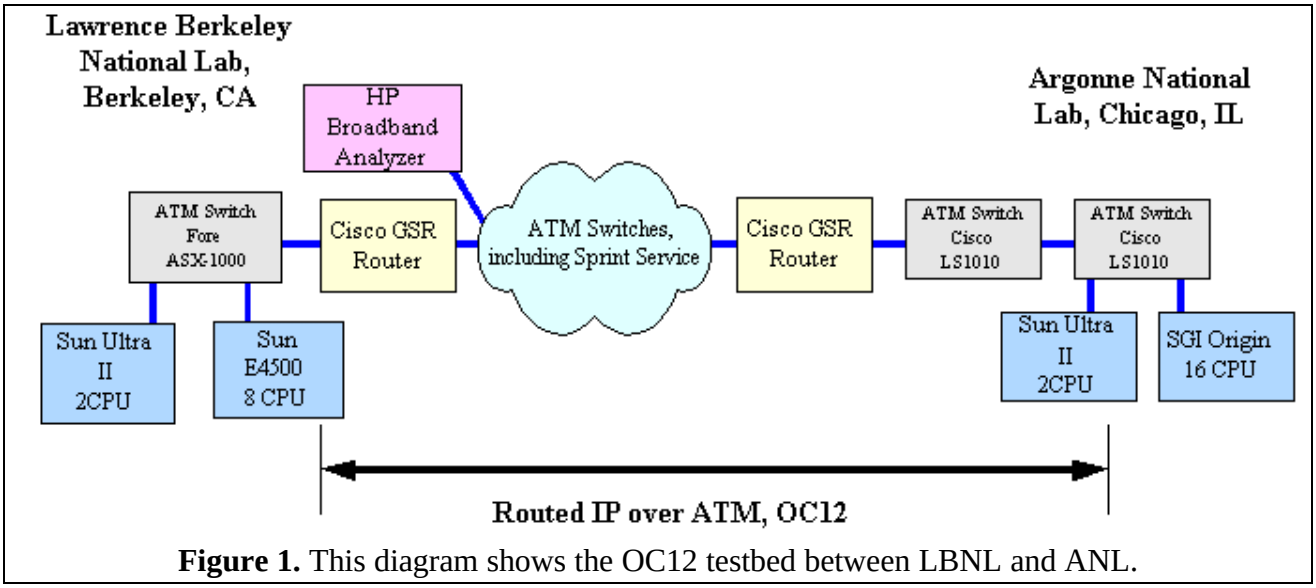

Figure 1. This diagram shows the OC12 testbed between LBNL and ANL.

1.2 Test Scenarios and Tools Used

In order to evaluate the link a HP Broadband Analyzer was used at the ATM level. TTCP was used for end-to-end TCP/IP and UDP/IP performance testing.

1.2.1 HP Analyzer for ATM Level Testing

The HP analyzer was used to test basic throughput at the ATM level by putting a loopback at the remote end of the link on an ATM switch and sending cells at OC12 rates. Subsequent loopback tests were done through the GSR router (see Figure 2). These particular tests were done at the ATM level only and not at the IP level. Therefore, the link loopback used on the Cisco GSR precluded any testing of the ATM policing parameters of the router at this step.

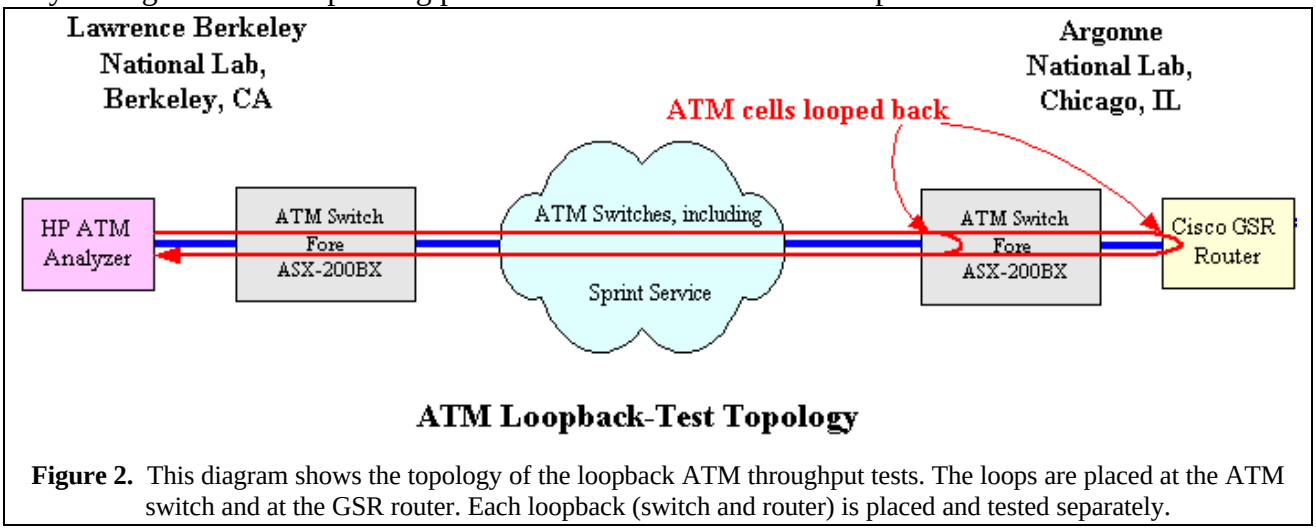

Figure 2. This diagram shows the topology of the loopback ATM throughput tests. The loops are placed at the ATM switch and at the GSR router. Each loopback (switch and router) is placed and tested separately.

The HP analyzer was used for ATM Generic Cell Rate Algorithm (GCRA) compliance testing of the Cisco, where the ATM policing parameters of the router was specifically tested [GCRA]. The GSR router has policing parameters that are configured per virtual circuit, and these must work as configured in order to pass policing algorithms in subsequent hops through an ATM policed network. In this test, the analyzer looked at the cell traffic generated by the router to see if it would adhere to cell policing algorithms. IP data traffic was generated with flood pings directed towards a recipient router. The data was duplicated at a switch and passed to the analyzer for study (see Figure 3).

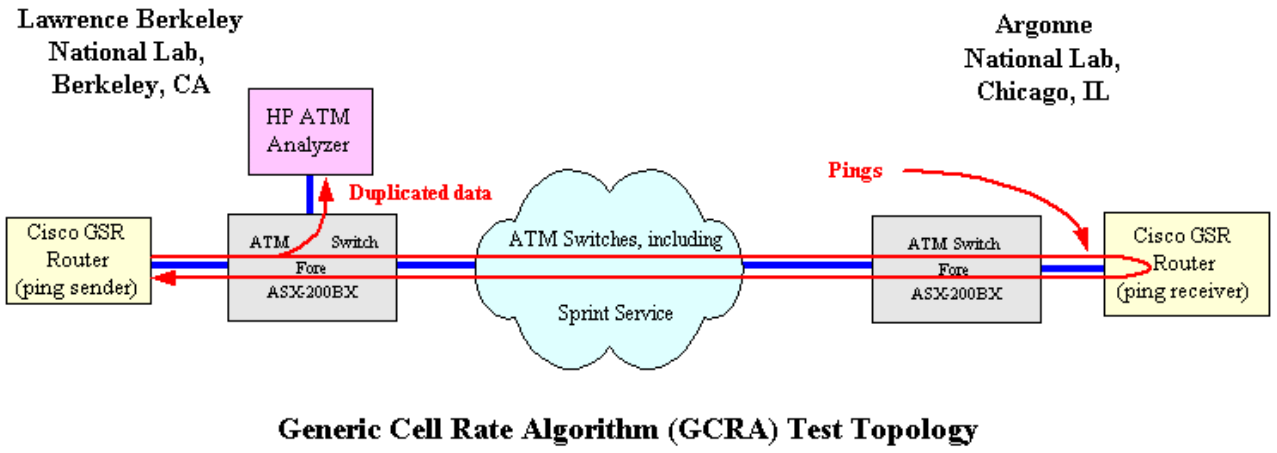

Figure 3. This diagram shows the path of the pings and the duplicated data sent to the analyzer for GCRA compliance testing.

The parameters evaluated were Peak Cell Rate (PCR), Sustained Cell Rate (SCR), and Maximum Burst Size (MBS).

1.2.2 TTCP for End-to-End TCP Testing

End-to-end performance testing was done using TTCP with large Maximum Transmission Units (MTUs). This was done with both TCP and UDP. While TCP performance issues are of more interest, the UDP and HP analyzer helped provide a baseline on what throughput to expect.

The first task was to tune the hosts. The MTUs used were approximately 9K. For TCP, the maximum send/receive buffers were set to approximately 4MB and the maximum congestion window (cwnd) was set to approximately 2MB.

2.0 Results

2.0.1 Theoretical Maximum Throughput Rates

In order to analyze the results, it is important to look at the protocol overhead for each layer (see Figure 4). The OC12 Sonet framing overhead is approximately 3.704 %:

622.08 Mbps – (3.704 % of 622.08 Mbps) = 622.08 Mbps – 23.04 Mbps
= 599.04 Mbps

The ATM overhead is 5 bytes per 53-byte cell, or 9.43 %:

599.04 Mbps – (9.43 % of 599.04 Mbps) = 599.04 Mbps – 56.49 Mbps
= 542.55 Mbps

The ATM Adaptation Layer 5 (AAL5), SubNetwork Attachment Point (SNAP) in this case, is 16 bytes per PDU [JH93]. Given that the MTU was near 9180 bytes, the AAL5-SNAP is less than .09 % overhead leaving the IP layer approximately 542.06 Mbps.

Layer:	OC12	Sonet	ATM	Adaptation	IP
Overhead:	0%	3.704%	9.43%	<.09% for 9180 PDU	<.22% for 9172 PDU
Bandwidth available after overhead:	622.08 Mbps	599.04 Mbps	542.55 Mbps	542.06 Mbps	540.87 Mbps

Figure 4. This shows the theoretical bandwidth available after protocol overhead.

2.0.1 HP Analyzer Throughput Testing

The rates achieved with these ATM level tests were approximately 572 Mbps both with the loopback on the ATM switch and on the router (see Figure 2). The throughput reported by the analyzer included the ATM overhead. This leaves approximately 27 Mbps unaccounted for since theoretically 599.04 Mbps should have been available to the ATM level (after Sonet framing overhead). The bandwidth provider, Sprint, informed us that this was due to sharing the OC12 link with other customers. Therefore the IP layer had approximately 527 Mbps available to it compared to the theoretical rate of approximately 542 Mbps.

2.0.2 HP Analyzer GCRA Compliance Testing of the CISCO GSR

Initial GCRA tests (see Figure 3) failed. A problem was determined with how Cisco implemented SCR and MBS. Cisco has since found and corrected the problem. GCRA tests now pass.

2.0.3 Local Baseline End-to-End TCP Throughput

TCP performance was measured between two local hosts in order to get a baseline and see what the limit was host-to-host with no delay to speak of, as well as less router and switch hops. A test was done between two hosts connected via OC12 ATM interfaces traversing a Fore ATM switch. A Cisco GSR router was added to the path in order to determine the impact of adding an IP routed hop (see Figure 5).

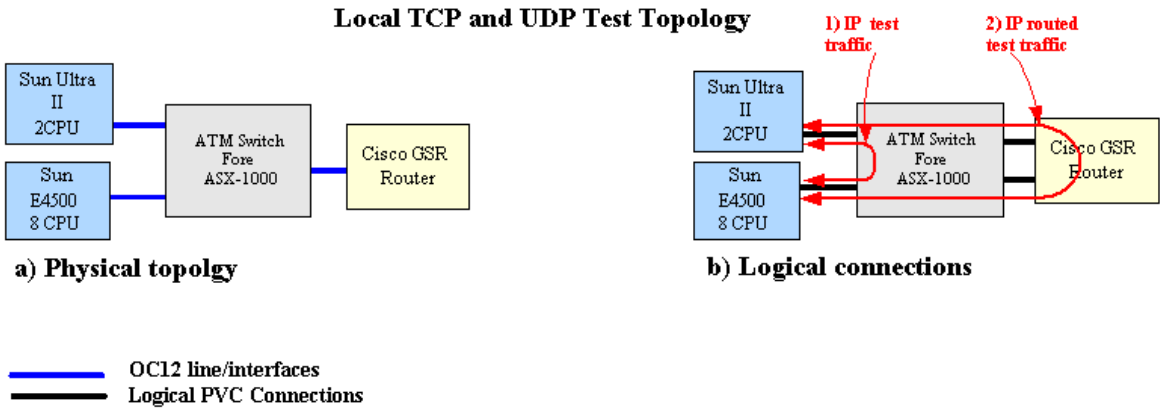

Figure 5. These diagrams show the local loopback test. Diagram **a)** on the left shows the physical components. Diagram **b)** on the right shows the logical VP/VC paths of the host-to-host IP traffic. Test 1) traverses the path host-switch-host and test 2) traverses the path of host-switch-router-switch-host.

The maximum IP throughput rate was 515 Mbps. This was using both TCP and UDP, as well as with and without the router in the path. It appeared that the router was not the bottleneck, leaving the host and ATM switch interfaces as possible suspects, with the hosts being the most likely candidates. This is because it was shown with the HP analyzer that these ATM switches could at least drive 572 Mbps.

2.0.4 End-to-End TCP Throughput

TCP performance is probably the most interesting part of our results. We had a unique situation where there was only one TCP session trying to use the full capacity of the pipe. The rates initially varied over a wide range of 150 – 300 Mbps.

It was then determined that over a TCP session that lasted 3-5 minutes there were 3-4 cell drops reported as interface CRC errors on the Cisco router. The cell drops caused the router to be unable to reconstitute the PDU from the stream of cells received. Therefore, each drop of 48 bytes of payload resulted in an entire PDU being dropped (in this case a packet of near 9180 bytes).

In looking at the TCP traces it appeared that the transfer was going into slow start after retransmission timeout. At this point we installed TCP Selective Acknowledgment (SACK) [MMF96], implemented according to RFC 2018, on the hosts [SACK]. Performance improved and TCP rates ranging from 300 – 400 Mbps were obtained. On several occasions that maximum throughput rate jumped to 450-480 Mbps.

The amount of packet loss due to errors was extremely small at less than .01 %, yet the difference in TCP performance with SACK (400 Mbps maximum) versus without SACK (300 Mbps maximum) was dramatic. Table 1 shows preliminary test results, where the network was not loaded with additional production IP traffic (other than the “lost” 27 Mbps). Table 2 shows the results of a more controlled set of tests, where SACK was in use, the network was being shared with other production traffic, and the cell drop problem had re-surfaced.

TCP/IP TEST	Throughput Range Mbps
Local Loopback through GSR Router	400 – 513
LBNL to ANL no SACK, with cell loss	150 – 300
LBNL to ANL with SACK, with cell loss	300 – 400

Table 1. This table shows a summary of throughput rates attained during preliminary tests where the network was not loaded with other production traffic.

TCP/IP TEST	Min Mbps	Max Mbps	Average Mbps	Std
ANL to LBNL 10 GB file	278	393	346	36.69
LBNL to ANL 10 GB file	285	352	352	23.59

Table 2. This table shows a summary of throughput rates for a 10GB file transfer where the network had other production traffic, there were cell drops and SACK was used. This was a more formal set of tests than shown in Table 1. Note that the slower rates are most likely due to the network being loaded with other traffic, as well as continued short bursts of cell loss.

3.0 Issues

Several issues were illuminated during these tests and are discussed below.

Cell Drops

Tracking down cell losses is very difficult. The switches along the path were controlled by two administrative domains, where one domain was accessible and one was not. For the accessible domain, after much painstaking work, a problem was finally identified and corrected with one of the ATM OC12 interface cards. For the administrative domain over which there was no control, coordination with the carrier is required in order to track down cell losses.

Given that the Cisco GSR routers passed GCRA compliance testing, it was assumed that policing parameters set inside the carrier switches (according to contractual agreements) *should not* be the cause of cell drops. However, in particular when ATM services were first being deployed, policing problems were common, where the bandwidth available was not in compliance with contractual agreements. This was due to switch algorithm problems as well as human configuration errors. In order to identify cell drops within another domain, the customer must rely on the administrator of that domain to run tests (in some cases intrusive tests) in order to identify problems.

TCP Behavior at High Speeds

When sending TCP at OC12 rates, TCP congestion avoidance [JK90] and slow start algorithms [SR97] can greatly impact the overall throughput. Packet drops are assumed to be due to congestion (hence the algorithm term “congestion avoidance”). In our tests the drops were caused by problems on the OC12 line card. The congestion avoidance algorithm was not helpful in this case. In part, SACK alleviates this problem by avoiding retransmission-timeout followed by slow start, which can increase the throughput rate significantly. There are some TCP enhancements being worked on that may help even more. For example, TCP Vegas [BP95] has additional bandwidth estimation capabilities that could help TCP recover from these errors more quickly.

4.0 SACK Improvements and Why

The TCP “fast-retransmit” algorithm [SR97] will quickly recover from one packet loss in a given congestion window, but not more than one packet loss. The TCP SACK extension was designed to avoid waiting for the TCP retransmission timer in the situation where there is more than one packet loss within the same congestion windows. On this network, we typically see a burst of several errors together. Because of this, SACK will recover from these errors much better than non-SACK TCP implementations, as is shown in the tables above.

5.0 Effects of TCP Slow Start

For large bandwidth-delay product networks such as this, it takes quite a while for the TCP slow start algorithm to fully open up the congestion windows. Figure 6 shows a TCP trace of the window opening up. For this network, which has a MTU of 9180 bytes, the round trip time is 45 ms for a 9 KB packet. We can see from the figure that it takes a total of 12 round trip times to fully open up the congestion window.

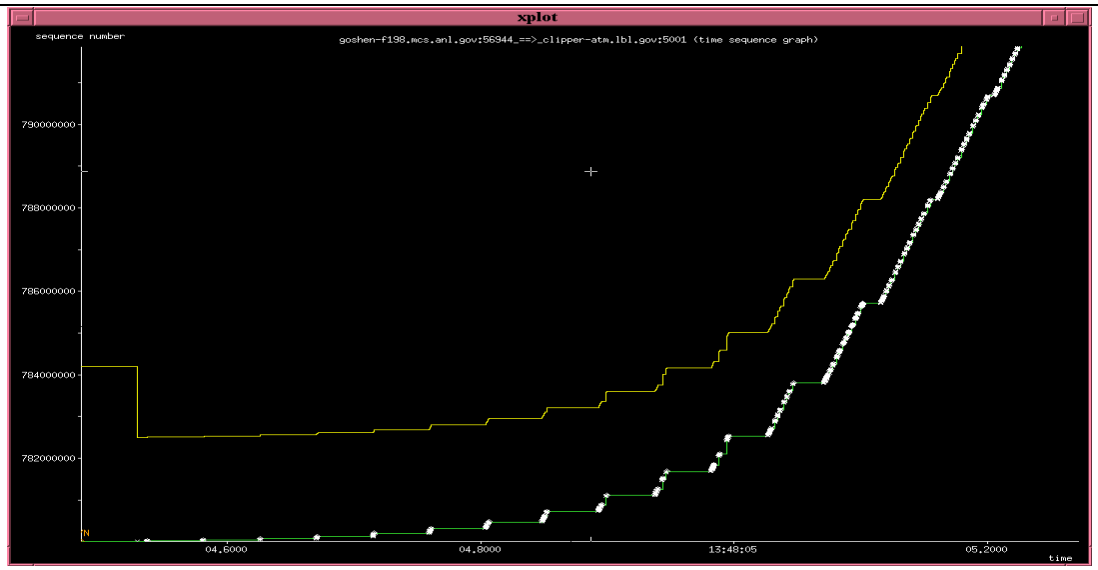

Figure 6. This diagram shows a TCP trace during initial slow start.

This seems like quite a bit of wasted time, but in fact, for large bulk data transfers where one would be concerned about this waste, the percent wasted is quite small. The theoretical rates are estimated along with the percentage of bandwidth “lost” due to slow start are shown in Table 3. This table shows that for 10 GB files the speedup is only .3 %, but for 100 MB files the speedup would be 23 % (speedup is computed by the formula defined in [THO96]). However one can envision a scenario where one might be using http to transfer lots of 100 MB files over a dedicated channel where congestion is not an issue. In this case, the current slow start mechanism is very inefficient. Methods for saving and reusing the previous window size, such as that suggested by slow start restart [VH97] or TCP Fast Start [PK98], would be very useful.

File Size	Minimum TCP/IP Transfer Time	Slow start "waste" time	Speed Up
10 GB	178 sec	.54 sec	.30%
1 GB	18.2 sec	.54 sec	2.9%
100 MB	2.31 sec	.54 sec	23.4%

Table 3. This table shows the of gain percentage for a file transfer's speed if slow start was eliminated. The larger the file, the less of an the impact slow start had.

6.0 History of TCP Enhancements for Large Bandwidth-Delay Networks

Over the past ten years there have been a number of enhancements to the TCP protocol and TCP implementation specifications which have helped improve TCP throughput over large bandwidth-delay product networks. In addition to TCP enhancements, improvements in hosts, switches and IP routers have helped as well. In this section we review the most significant of these improvements.

The first time the original TCP specification was found to be lacking was when the Pittsburgh Supercomputer Center (PSC) tried to run TCP over a 800 Mbps HIPPI network between 2 Cray computers in 1990. The original TCP receive buffers where way too small for a network that fast, even if the latency is low. This lead to Jacobson and Borman to develop the TCP window-scale option [JBB92], which allowed then to get 780 Mbps of TCP throughput between the Cray systems.

The next major obstacle to TCP performance was discovered by the Gigabit Testbeds [GIGA]. Several groups from the Casa, Aurora, VistaNet, Blanca, and MAGIC testbeds were trying to get decent performance across wide area OC-3 links, and found they could only get TCP speeds of 30-40 Mbps out of the possible 131Mbps. It was determined that this was mainly due to the memory to memory copy speed of that generation of workstations. This discovery lead to several enhancements, including zero-copy TCP implementations, hardware support for computing the TCP checksum, and improved memory bus architectures by the workstation vendors.

The most important of these was zero-copy TCP. At this time most TCP implementations did a memory to memory copy while reconstructing the TCP packet from a series of IP packets. Jacobson showed that this wasn't necessary, and that a much more efficient implementation could be done by eliminating "layering". He also showed that the checksum could be computed while the data was being copied into memory from the device driver, effectively making the checksum operation free [JA93]. Eventually all TCP implementations started doing this [e.g.: CJ96]. About this same time, workstation vendors started to improve the memory bus, starting with Digital (Dec Alpha model 3000). By 1993, several Gigabit testbeds reported TCP throughputs in and out of a single host at 130Mbits for a single OC-3, and 200Mbits over multiple OC-3's.

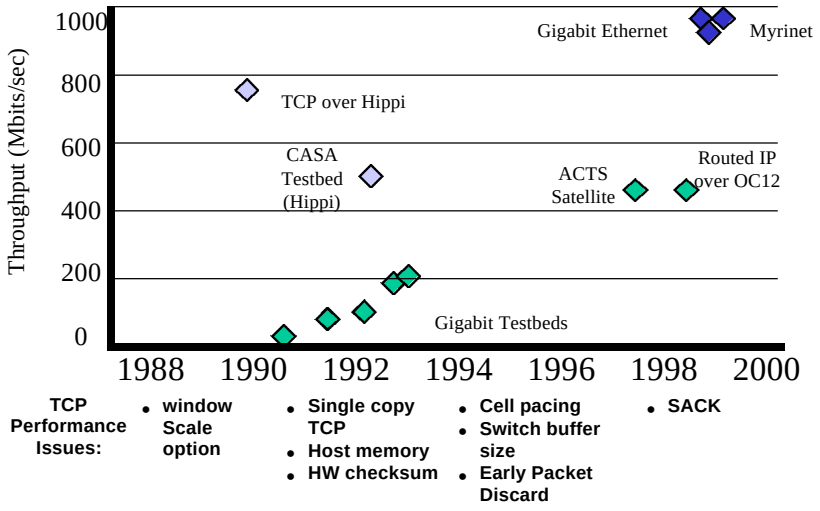

Figure 7. This diagram shows the TCP/IP throughput rates achieved over high speed networks.

Now that the host was no longer the bottleneck to wide-area TCP performance, the next set of obstacles was ATM switches and IP routers. The first generation of ATM switches had a very small amount of buffering. For example, the FORE ASX-100 had only 9K of buffers per output port. Multiple TCP streams, or even a single bursty host, could easily overflow the buffer, causing lost ATM cells and hence lost IP packets [TJH96]. Several enhancements addressed this problem, including the addition of the ability to do cell pacing in the ATM device drivers, larger switch buffers, and adding "early packet discard" to the switches. With early packet discard, the ATM switch is smart enough to not just drop one cell, but to drop all ATM cells in an IP packet, thereby avoid the extra load on the network of data that will have to be retransmitted later. [RF95]

The next TCP enhancement that helps improve high-speed WAN performance was not actually designed for this purpose, but seems to help quite a bit. The TCP Selective Acknowledgment option, or SACK, was designed to avoid having to wait for the TCP retransmission timer to go off for cases where two packets in the same congestion window are lost [MMF96]. Although this option was designed mainly to help improve performance on heavily congested Internet links, it also helps quite a bit on lightly loaded high-speed wide area links, as show above. This is due to the nature of errors in this environment, where often a group of errors happen together, for reasons we do not yet understand. NASA also reported a large performance increase using TCP with SACK over OC-12 satellite links [ACTS] [CLF98].

Currently there appears to be no particular limitation to achieving 1 Gigabit/second of TCP throughput. Alteon has reported gigabit Ethernet speeds of TCP throughput of 990 Mbps with Sun hardware with Solaris, and 940 Mbps using Compaq hardware with Windows NT, when using gigabit Ethernet "jumbo frames" of 9KB [GIGA]. Also, Duke University has reported TCP speeds of 952 Mbps over Myrinet [MYRI]. However host memory bus bandwidth will almost certainly become an issue again when OC-48 (2.4 Gbps) interfaces become available later this year.

Figure 7 shows some of the more significant milestones in terms of TCP performance, and shows what TCP, switch, or router enhancements occurred at that time to help provide those that

performance. Most of these results are from the Gigabit Testbed Initiative final report [CF94] [GT96]. Figure 8 shows the same information in packets/second, and shows a fairly linear increase in packets / second over the past 10 years.

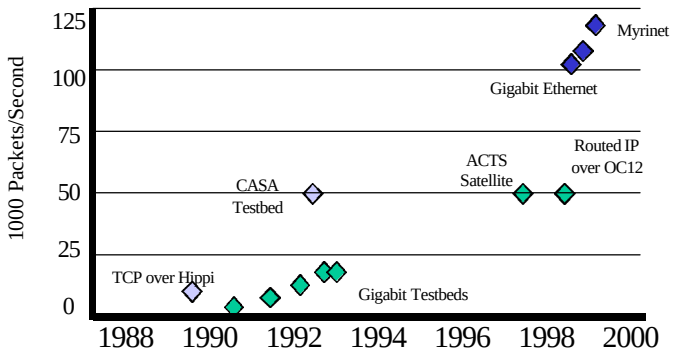

Figure 8. This diagram shows the packets/second rates achieved over high speed networks.

7.0 Future Work

We are looking at tools to better evaluate our ATM service. Both in finding problems such as cell drops due to interface errors, and cell drops due to incorrect policing parameter configurations or implementations of policing parameters in the switches and routers. We still need to determine why there were consecutive cell drops.

During the local loopback tests we found that the maximum throughput rates host-to-host were 515 Mbps at the IP layer. We would like to understand why the hosts achieved only 300-400 Mbps were not able to reach the theoretical rate (at the IP level) of 542.06 Mbps. And further, why there was such a wide range of throughput rates.

One obstacle in the testing was that the environment was continuously changing. For example, the network was relatively unloaded and then became more loaded with bursty IP traffic once it went into a production mode. This made it difficult to track test results. In the future we plan to run more tests over longer periods of time, gather more information including monitoring the production load to track potential impacts it may have on the tests.

Finally, we would like to do additional analysis of the TCP performance over the OC12 with different variations of TCP to evaluate packet loss handling and slow start.

8.0 Conclusions

When dealing with OC12 line rates, even very small packet loss can have a dramatic impact on TCP performance. With a packet loss of less than .01 %, the throughput rate ranged from 150-300 Mbps. With SACK the throughput increased to a range of 340-400 Mbps.

Finding and correcting cell loss, in particular when the cell loss is small, proved to be very difficult. Given the impact of these cell losses, it is desirable to find and correct them.

We also determined that for large bulk data transfers near OC12 rates, the impact of slow start was negligible.

9.0 Acknowledgements

We would like to thank the following people from LBNL who helped gather test data: Jin Guojun, Jason Lee, and Kevin Oberman. And Jim Gagliardi from LBNL for his diligent work in tracking down the cell drop problem. From ANL, Linda Winkler and Bill Nickless were most helpful in coordinating the ANL side of the testbed.

10.0 References

[ACTS] NASA ACTS Satellite, http://mrpink.grc.nasa.gov/gsnhome.html

[BJZ90] Braden, R., Jacobson, V., and Zhang, L., "TCP Extension for High-Speed Paths", October 1990. RFC 1185.

[BP95] Brakmo, L. and Peterson, L. "TCP Vegas: End to end congestion avoidance on a global internet", IEEE Journal of Selected Areas in Communication, 13(8):1465-1480, October 1995.

[CLF98] C.P. Charalambous, G.Y. Lazarou, V.S. Frost, J. Evans, R. Jonkman, "Experimental and Simulation Performance Results of TCP/IP over High-Speed ATM over ACTS", Proceedings of 1996 IEEE International Conference on Communications, Atlanta, Georgia, June 1998.

[CF94] Chinoy, B. and Fall, K. "TCP/IP performance in the CASA Gigabit Network", Proceedings of the Usenix Symposium on High-Speed Networking, Aug 1994.

[CJ96] Chu J.H., "Zero-Copy TCP in Solaris", Proceeding of Usenix 1996 Annual Technical Conference, http://www.usenix.org/publications/library/proceedings/sd96/chu.html.

[GCRA] The Generic Cell Rate Algorithm (GCRA) traffic scheduling defined by the ATM forum in the Traffic Management Specification #af-tm-0056.000.

[GIGA] Gigabit Ethernet Performace: http://colo.tntmedia.com/~alteon/products/perfcharts.html

[GT96] The Gigabit Testbed Initiative Final Report, December 1996; http://www.cnri.reston.va.us/gigafr/

[JA93] Jacobson, V., "Some Design Issues for High-speed Networks" (viewgraphs). Networkshop '93, Melbourne, Australia, November 30, 1993. ftp://ftp.ee.lbl.gov/talks/vj-nws93-1.ps.Z

[JBB92] Jacobson, V. , Braden, R., and Borman, D., "TCP Extensions for High Performance", May 1992. RFC 1323.

[JH93] Heinanen, J., "Multiprotocol Encapsulation over ATM Adaptation Layer 5", July 1993. RFC 1483.

[JK90] V. Jacobson and M. Karels. "Congestion Avoidance and Control". *ACM SIGCOMM Computer Communication Review*, August 1990.

[MMF96] Mathis, M., Mahdavi, J., Floyd, S., Romanow, A., "TCP Selective Acknowledgment Options", October 1996. RFC 2018.

[MYRI] Myrinet Performance: http://www.myri.com/myrinet/performance/index.html

[PK98] Padmanabhan, V., and Katz, R., "TCP Fast Start: A Technique For Speeding Up Web Transfers", Proc. IEEE Globecom '98 Internet Mini-Conference, Sydney, Australia, November 1998. (http://www.cs.berkeley.edu/~padmanab/papers/gi98.ps)

[RF95] Romanow, A., and Floyd, S., "Dynamics of TCP Traffic over ATM Networks", IEEE JSAC, V. 13 N. 4, May 1995, p. 633-641.

[SACK] Experimental TCP Selective Acknowledgment Implementations Links: http://www.psc.edu/networking/all_sack.html

[SR97] Stevens, W.R. “TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms”, January 1997. RFC 2001.

[TJH96] Tierney, B., Johnston, W., Hoo, G., and Lee J., “Performance Analysis in High-Speed Wide Area ATM Networks: Top-to-bottom end-to-end Monitoring”, IEEE Network, Vol. 10, no. 3, May/June 1996, LBL-38246.

[THO96] Touch, J., Heidemann, J., Obraczka, K., “Analysis of HTTP Performance”, USC / Information Sciences Institute white paper, http://www.isi.edu/lsam/publications/http-perf/

[VH97] Visweswaraiah, V. and Heidemann, J, “Improving Restart of Idle TCP Connections”, Technical Report 97-661, University of Southern California, November, 1997.