On the Impact of BER on Realistic TCP Traffic in Satellite Networks

Priya Narasimhan[*]   **Hans Kruse**[†]

Shawn Ostermann[‡]   **Mark Allman**[§]

TR-04-005

November 2004

Abstract

There are many factors governing the performance of TCP-based applications traversing satellite channels. The end-to-end performance of TCP is known to be degraded by the delay, noise and asymmetry inherent in geosynchronous systems. This result has been largely based on experiments that evaluate the performance of TCP in single flow tests. While single flow tests are useful for deriving information on the theoretical behavior of TCP and allow for easy diagnosis of problems, they do not represent a broad range of realistic situations and therefore cannot be used to authoritatively comment on performance issues. The experiments discussed in this report test TCP's performance in a more dynamic environment with competing traffic flows from hundreds of TCP connections running simultaneously across the satellite channel. Another aspect we investigate is TCP's reaction to bit errors on satellite channels. TCP interprets loss as a sign of network congestion. This causes TCP to reduce its transmission rate leading to reduced performance when loss is due to corruption. We allowed the bit error rate on our satellite channel to vary and tested the performance of TCP as a function of these bit error rates. Our results show that the average performance of TCP on satellite channels is good even under conditions of loss as high as bit error rates of 10^{-5}.

[*]Network General, `priya.narasimhan@networkgeneral.com`
[†]Ohio UNiversity, `hkruse1@ohiou.edu`
[‡]Ohio University, `ostermann@ohiou.edu`
[§]ICSI / ICIR, `mallman@icir.org`

1 Introduction

The use of satellite networks as a part of the Internet backbone dates back almost twenty five years [Hen00]. Since then the market for satellite communications services and technology has grown rapidly. A standardized suite of data communication protocol standards would be beneficial to both the military and civilian satellite communication user communities [DMT96]. In developing a standard protocol suite for use in space, it is natural to draw from the long standing, successful and robust TCP/IP suite.

A large amount of work has been carried out to determine the impact of a geosynchronous satellite in the network path on the performance of TCP [AKO00, ADG$^+$00]. Allman et al. [AGS99] describes in detail the impact satellite links have on TCP, and specifically, how such channels degrade the performance of TCP. Caceres and Iftode [CI95] analyzes the performance of TCP on networks that include wireless links, which inherently suffer from delays and packet losses. Their experiments studied the performance of a single steady state TCP connection in a wireless networking testbed. [CI95] concludes that TCP reacts to delays and losses by abruptly slowing its transmissions. All these results are based on experiments conducted to measure the performance of a single, steady-state TCP flow, and have indicated that TCP performance begins to degrade noticeably for bit error rates above 10^{-8} on a T1 link [KOA00]. The results show exponential decline with error rates above this level, with throughput cut roughly in half by the time the bit error rate reaches 10^{-7}.

As outlined in [AF99], results based on single flow tests cannot be used to illustrate the performance of TCP as it is used in real networks. [AF99] recommends that TCP's performance should be tested in dynamic environments with competing traffic flows. We investigate TCP's performance, when a large number of connections are active in parallel across a satellite channel and subjected to slowly varying error rate conditions on the link. This multi-flow traffic is similar to what one would expect on a real network, and should be the condition under which performance of TCP is evaluated.

The remainder of this paper is organized as follows. Section 2 details the experimental setup and methodology used in our study. Section 3 outlines the results of our performance analysis. Finally, Section 4 gives our conclusions.
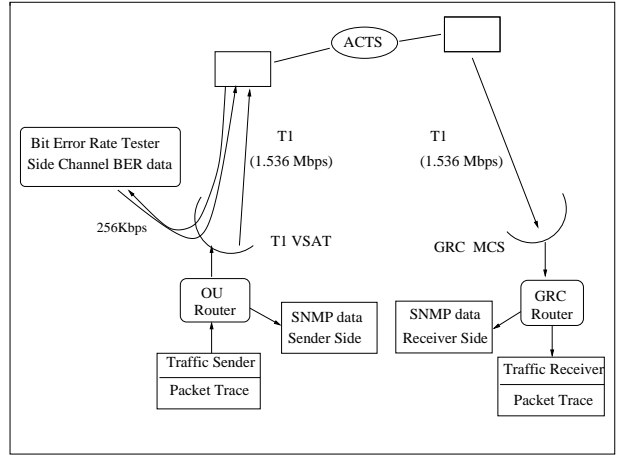

Figure 1: Experiment Setup

2 Experiment Setup

To evaluate realistic TCP traffic, we setup experiments across NASA's Advanced Communications Technology Satellite (ACTS) [NAS]. The network layout consisted of workstations using the NetBSD 1.3 Unix operating system located on the Ohio University (OU) campus in Athens, OH, as well as at the NASA Glenn Research Center (GRC) in Cleveland, OH. The workstations at OU transmit and receive using an ACTS T1 VSAT terminal (with a 1.2m reflector) across the ACTS satellite to the Master Ground Station at GRC as shown in Figure 1.

Geostationary satellites tend to *drift* due to the gravitational pull of the sun, other objects in the solar system and the uneven distribution of land mass on the surface of the Earth [McL00]. To counteract these forces, the satellite must be fitted with some mechanism to move it back into position when it drifts. This process of maintaining the satellite at proper position and attitude is called stationkeeping and utilizes the bulk of the fuel on a spacecraft. To conserve fuel, north-south stationkeeping was discontinued on ACTS in July 1998 causing the satellite's orbit to become inclined, i.e. it oscillated on its north-south axis. Note: East-West stationkeeping is mandatory, but uses much less fuel. The Master Ground Station at GRC and the T1 VSAT at OU can both track the satellite's movement so that their send and receive antennas are always aligned. However, we disabled tracking on the T1 VSAT antenna (at OU) to produce varying bit error rates on the link.

The Master Ground Station at GRC uses *Forward Error Correction* (FEC) [Fle99] to clean up the channel. FEC is important for a satellite operating in the Ka-band because the wavelengths in the Ka-band are

close to the size of raindrops, leading to a scattering of incident energy, causing *rain fades* [Bau97]. ACTS uses convolution coding [Fle99] for FEC and is setup such that the user can specify the desired level of FEC. For our experiments, we set the threshold for FEC at 10^{-3}, i.e., the link would tolerate bit errors of up-to 10^{-3} before employing FEC.

The data rate across this link was standard T1 rate (1.536 Mbps). The round-trip time across the network was approximately 560 msec. To generate time dependent Bit Error Rate (BER) patterns, the VSAT's tracking was moved ahead and off-center of the satellite central beam and locked in position. This mis-pointing was adjusted to produce BERs of about 10^{-3} on the OU-to-ACTS link. As the satellite moved in its inclined orbit towards the VSAT antenna position, the BER gradually decreased, and as it moved away, the BER gradually increased. Due to the nature of our experimental setup, link fades primarily affected the T1 VSAT to ACTS transmit link. The Master Ground Station at GRC tracked the satellite, so the ACTS to GRC link would be relatively error free. Any packets transmitted by GRC would reach OU after a hop through the satellite. Since the satellite's antenna transmits at a greater power than the T1 VSAT, the packets traveling from ACTS to the VSAT are received with a low BER. This means that bit errors appear almost exclusively on the packets traveling from OU to GRC, while the packets that traveled from GRC to OU will not be greatly affected by BERs. The test runs were set up to last roughly 3 hours (enough time to observe a wide range of BERs).

To measure the BER on the satellite channel, a separate 256 Kbps loopback circuit was set up between OU and the satellite. The BER on this link was measured using an HP Data Communications Tester. This tester sends random bit patterns to the satellite and computes the combined BER on the uplink and downlink. Since the downlink is expected to be relatively error free, this BER was assumed to represent the bit error rate on the uplink. It is important to re-iterate that the channel used to measure the BER was different from the channel used to transfer the TCP traffic. This would imply that the BER measured on the side channel may be different from the traffic channel. We believe that this difference would be negligible over relatively long time scales. The total bit errors were recorded approximately every 15 seconds.

The routers used in the experiments were all Cisco 2500 series and were set up with differing queue lengths for different runs. They were queried periodically using the Simple Network Management Protocol (SNMP) [CFSD88] to gather data such as bytes transferred, bit errors observed and packets dropped on each of the serial and Ethernet interfaces.

TCP traffic from both OU and GRC was generated using *TrafGen* [Hel98], a traffic generation application that generates multiple simultaneous TCP flows in patterns similar to those observed on real networks. *TrafGen* generates controlled amounts of TCP/IP traffic from selectable profiles, between the traffic server and the client. Once *TrafGen* has been compiled to emulate a certain traffic pattern [KAG+99], the amount of traffic generated can be controlled with a single run-time parameter referred to as the *Big Red Knob* (BRK). The value of the BRK is applied to determine the inter-arrival time between connections. A smaller inter-arrival time would create a greater number of connections per unit time, while a larger inter-arrival time would produce fewer connections per unit time. An important implication of using *TrafGen* is that although link conditions will affect the behavior of individual connections, *TrafGen*'s rate of creation of connections is unaffected by link conditions. While this aspect of *TrafGen* may not necessarily completely mirror reality, we believe the traffic pattern generated is realistic enough for our purposes since we are not attempting to derive information about application layer or user-perceived performance.

3 Performance Analysis

As sketched above, analyzing the performance of single flow TCP connections across the ACTS satellite is a first step in understanding TCP's dynamics in satellite-based networks. As a next step, we conducted experiments with a more realistic traffic pattern consisting of smaller, concurrent TCP flows. In this section we present the results of these experiments.

One of the tools we use to analyze the data is *tcptrace* [Ost97], a TCP packet dump file analysis tool. Using *tcptrace*, we produce several different types of output containing information about each connection observed, such as elapsed time, bytes transmitted and segments sent and received, retransmissions, round trip times, window advertisements, throughput and more. This output gives an overall view of the connections and their aggregate performance on the satellite link. We also look at the per connection view, by producing output based on the performance of each connection. We analyze the data in 5 minute intervals and plot the mean of every metric for each interval (unless otherwise noted).

We separate the packet trace files to allow analysis of the two directions of traffic independently. As explained in Section 2, the BER affects only the traffic from OU to GRC, while the return path is relatively error free. Since our objective is to analyze the effect
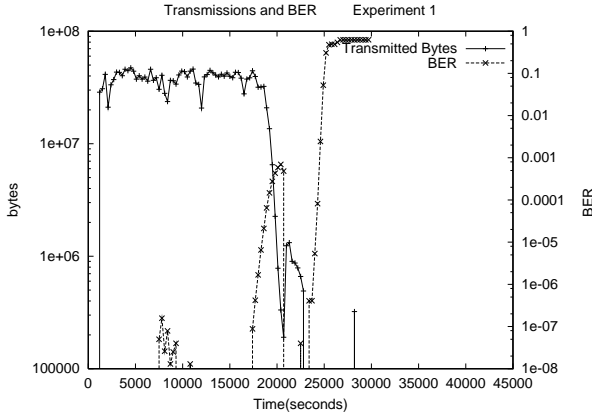
Figure 2: Transmitted Bytes and BER Over Time for Experiment 1

Figure 3: Mean Link Utilization for Experiment 1

of errors on TCP's performance, we separate the trace files into the two directions and all the data presented in this paper is from the direction in which the data packets, rather than the acknowledgments are affected by the BER.

During our analysis, we observed that at times a BER spike hits the link and completely halts the data transfer across the satellite. This causes some periods of relatively low throughput. Since we were not interested in looking at time periods with low traffic volume, we removed these sections from our analysis using *tcpslice* [Pax96].

Before we discuss the analysis of our data, an important point to emphasize is that we are analyzing artificially generated traffic. The traffic flows are created based on observing traffic on a real network and generating similar patterns. The amount of traffic produced depends on the BRK value. If the BRK value is greater than 1, the traffic load generated is lower than the observed traffic on the real network. When we talk about *link utilization* during a particular time interval, we are referring to the percentage of the total link capacity that is used by the total number of bytes per second flowing across the link. The number of bytes flowing across the link is affected by the maximum queue lengths set in the routers and the BRK values. Therefore, different experiments will show different link utilization because they were generated with different parameters. Given a traffic profile, we assume the link utilization achieved during the error free intervals as the best case. We then analyze the intervals with non-negligible BER based on their degradation from the error free case.

The results of our analysis are significant because they show the performance degradation observed during the flow of competing traffic from many TCP connections is not as severe as predicted for a single flow
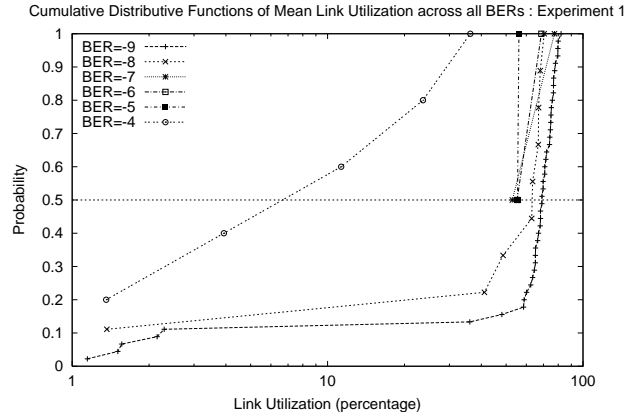
TCP connection. We observed the average link utilization used by TCP traffic begins to degrade significantly only at BERs of above 10^{-5}. We studied the link utilization and aggregate throughput achieved by the all the connections across the range of the BERs. This aspect is discussed in Subsection 3.1. Subsection 3.2 discusses the per connection performance across the range of BERs.

3.1   Macroscopic Analysis

Figure 2 shows the relation between the transmitted bytes and BER over time for experiment 1. We observe that as the BER rises (between 17,000 and 20,000 seconds) the total number of bytes flowing across the ACTS link is largely unaffected. However, when the BER increases to about 10^{-3}, the throughput across the link drops dramatically. In this experiment, most of the connections were never able to recover from the high BER on the link and could not recover even after the channel was cleaned up with FEC. The sudden drop in the BER at about 21,000 seconds is due to FEC taking effect. Figure 2 shows that when the link is error free between 21,000 and 24,000 seconds, some of the connections do recover and try to send data again. However, the link finally dies at about 24,000 seconds when the BER approaches 1.

Figure 3 shows the cumulative distribution functions of the average link utilization (per 5 minute slice) for each BER observed, during experiment 1. The hairline at 50% represents the median. For example, reading off the intersection of the hairline and the $BER = -9$ line, we observe that when the BER on the link was 10^{-9}, half of the intervals recorded a link utilization below 70%. The 10^{-8} utilization follows closely with roughly 60%, while 10^{-7}, 10^{-6}, 10^{-5} all show a utilization of approximately 50%. Taking the BER of 10^{-9} as the error free case, we can observe
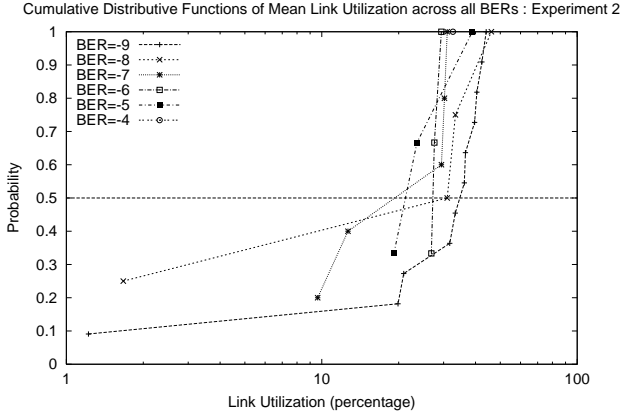
Figure 4: Mean Link Utilization for Experiment 2

Figure 5: Transmitted Bytes, BER and Router Drops Over Time for Experiment 2

that the median utilization at a BER of 10^{-5} shows a degradation of 29% ($\frac{70-50}{70}$ x 100) from the error free case. The maximum link utilization recorded (the value at probability 1) shows that the mean utilization per 5 minute interval in the error free case is about 80%, while that at a BER of 10^{-5} is 55%, which is roughly 31% ($\frac{80-55}{80}$ x 100) worse. This means that the average link utilization observed during BERs of 10^{-5} was never worse than 31% of the error free case.

The experiment shown in Figures 2 and 3 is one of the experiments where we observed expected patterns, i.e. as the BER increases, the link utilization decreases. However, this pattern was not observed in all the experiments. Some runs showed worse utilization during the error free case when compared to higher BERs. This could be because of the queue length setting on the routers. The smaller the queue length, the greater the probability that a packet sent to the router will be discarded (all other things being the same), leading to more dropped packets. This leads to retransmits in TCP, which results in reduction of the TCP congestion window and re-sending data. All of these factors contribute to lowering the link utilization. The following analyses deal with experiments where we observed unexpected patterns.

Figure 4 shows a utilization distribution for experiment 2. Analyzing this plot in a similar manner to the above, we observe that the median utilization in the error free case is 32%. The 10^{-8} case follows with a utilization of 30%. Surprisingly, the 10^{-6} and 10^{-5} cases perform better than the 10^{-7} case, half the time, with the 10^{-7} recording a median link utilization of only 18%. When we look at the maximum link utilization, we observe that the 10^{-9} performs slightly worse than 10^{-8}. More significant however, is that the 10^{-5} case records a higher maximum utilization than either of the 10^{-6} or 10^{-7} cases, showing a degradation of
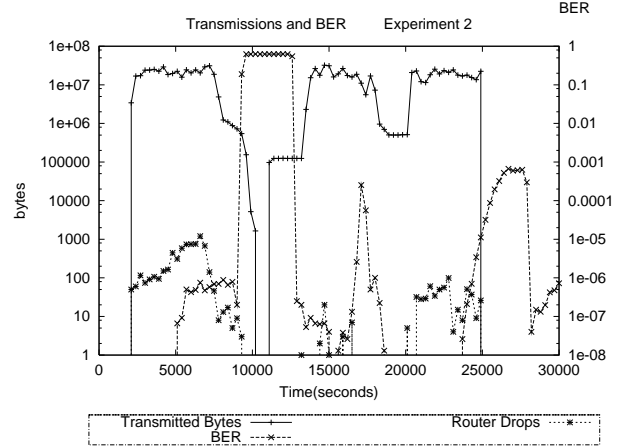
only 14% from the error free case. This unexpected pattern could be explained by looking at Figure 5. During this experiment there were a significant number of drops due to queue over-flow (i.e., congestion) during the low BER cases. We can observe that the number of corruption-based drops when experiencing a BER of 10^{-6} and 10^{-7} is high, while there were no router queue drops during BERs of 10^{-5}. These results may indicate that when the BER is "just right" the random corruption-based losses keep the TCP sources sending at a rate that does not lead to congestion of the link. This would seem to be a natural occurrence of the general mechanism that active queue management (e.g., RED [FJ93]) strives to provide.

The above analyses help us to conclude that the average link utilization by TCP, on lossy satellite channels, degrades by roughly 30% from the error free case for BERs of 10^{-5} and lower.

3.2 Microscopic Analysis

We now turn our attention to a more detailed analysis of the throughput attained by *each TCP connection* over the course of our experiments. Figure 6 shows the distribution of throughput for experiment 1 divided up based on the BER of the ACTS link during the connection. When the BER is 10^{-4}, TCP connections nearly always experience lower performance than connections at all other BERs. On median, connections attain 25% less throughput when operating at a BER of 10^{-4} when compared to the error-free case (10^{-9}). With the exception of the tail end of the 10^{-5} curve, TCP performance is not greatly effected by BER across the remaining BERs observed. This indicates that once enough statistical multiplexing occurs on the link, BERs do not have a large impact on
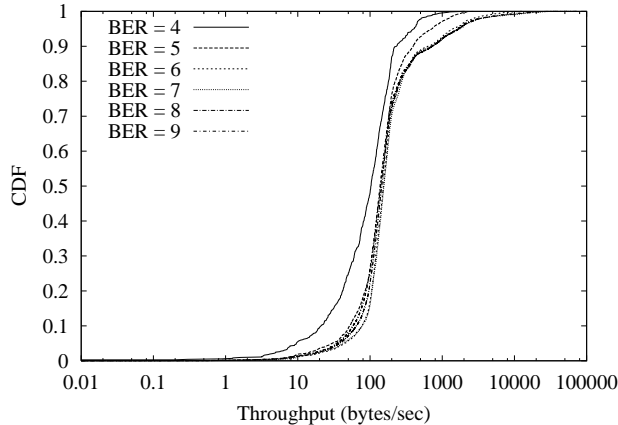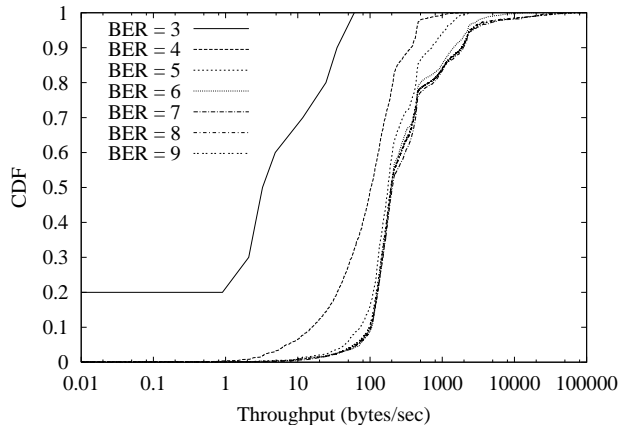
Figure 6: Throughput for Experiment 1

Figure 7: Throughput for Experiment 3

4   Conclusions

Our experiments have shown that aggregate performance of TCP across satellite links is much better than predicted by previous research consisting of only single TCP flows. Although an individual connection may notice a performance reduction on a satellite link due to non-negligible BER, when a realistic traffic pattern is used, the aggregate performance of all TCP connections is generally close to the maximum.

Allman et al. [AHKO97, KOA00] concludes that TCP's performance shows an exponential decline for error rates above 10^{-8} and that the throughput is cut roughly in half by the time the BER reaches 10^{-7}. However, our analysis has shown that when hundreds of connections are running in parallel across the link, the average throughput falls significantly only when the BER reaches a value of 10^{-5} or more. We observed that the link utilization degradation across the range of BERs from error free to 10^{-5} is 30% or less. We also observed that the average throughput recorded for all BERs in the range 10^{-9} to 10^{-5} varies significantly only in 20% of the connections. This suggests that in general, the aggregate TCP performance is largely invariant of BER patterns. The high BER conditions may hurt individual connections, but when a large number of parallel connections are involved, the aggregate performance of TCP is not greatly affected. This means that even though the BER does affect the performance of some connections across satellite channels, most of the connections are not affected adversely enough to cause significant performance degradation.

We also show that the distribution of per-connection throughput is largely independent of the error rate at BERs lower than 10^{-4}. Together with the above results, this indicates that higher BER links can be used to carry a large amount of realistic traffic without substantially reducing performance or user-perceived response time.

Finally, we note that the experiments discussed in this paper represent an initial foray into assessing the implications of statistical multiplexing in satellite networks. Continued work is required to further our understanding, develop models and ultimately design mitigations that can enable standard protocols to better utilize bandwidth across high-latency links.

Acknowledgments

specific TCP transfers.

Figure 7 shows the distribution of per-connection throughput for experiment 3 across all BERs observed. In this experiment, we operated with a BER of 10^{-3} at times. While the plot does not have a large number of data points for BERs of 10^{-3}, the points that are available suggest that this error rate has a serious impact on TCP performance. Again in this experiment there is a noticeable deviation in per-connection throughput when the BER is 10^{-4} and a slight deviation from the error free case at the tail end of the 10^{-5} distribution. When BER is less than 10^{-5}, the results suggest that errors are being spread across enough connections that the distribution of per-connection throughput remains roughly the same across all BERs observed during our test.

References

[ADG+00] Mark Allman, Spencer Dawkins, Dan Glover, Jim Griner, John Heidemann, Tom Henderson, Hans Kruse, Shawn Ostermann, Keith Scott, Jeff Semke, Joe Touch, and Diepchi Tran. Ongoing TCP Research Related to Satellites, February 2000. RFC 2760.

[AF99] Mark Allman and Aaron Falk. On the Effective Evaluation of TCP. *Computer Communication Review*, 29(5):59–70, October 1999.

[AGS99] Mark Allman, Dan Glover, and Luis Sanchez. Enhancing TCP Over Satellite Channels Using Standard Mechanisms, January 1999. RFC 2488, BCP 28.

[AHKO97] Mark Allman, Chris Hayes, Hans Kruse, and Shawn Ostermann. TCP Performance Over Satellite Links. In *Proceedings of the 5th International Conference on Telecommunication Systems*, pages 456–469, March 1997.

[AKO00] Mark Allman, Hans Kruse, and Shawn Ostermann. A History of the Improvement of Internet Protocols Over Satellites Using ACTS. In *Proceedings of ACTS Conference 2000*, May 2000.

[Bau97] Robert Bauer. Ka-Band Propagation Measurements: An Opportunity with the Advanced Communications Technology Satellite (ACTS). *Proceedings of the IEEE*, 85(6), 1997.

[CFSD88] J. Case, M. Fedor, M. Schoffstall, and J. Davin. A Simple Network Management Protocol, August 1988. RFC 1067.

[CI95] Ramon Caceres and Liviu Iftode. Improving the Performance of Reliable Transport Protocols in Mobile Computing Environments. *IEEE Journal on Selected Areas in Communications*, June 1995.

[DMT96] Robert Durst, Gregory Miller, and Eric Travis. TCP Extensions for Space Communications. In *ACM MobiCom*, November 1996.

[FJ93] Sally Floyd and Van Jacobson. Random Early Detection Gateways for Congestion Avoidance. *IEEE/ACM Transactions on Networking*, 1(4):397–413, August 1993.

[Fle99] Chip Flemming. A Tutorial on Convolutional Coding with Viterbi Decoding, 1999. http://pw1.netcom.com/~chip.f/viterbi/tutorial.html.

[Hel98] Eric Helvey. Trafgen: An Efficient Approach to Statistically Accurate Artificial Network Traffic Generation. Master's thesis, Ohio University, June 1998.

[Hen00] Tom Henderson. *Networking Over Next-Generation Satellite Systems*. Ph.D. thesis, University of California, Berkeley, 2000.

[KAG+99] Hans Kruse, Mark Allman, Jim Griner, Shawn Ostermann, and Eric Helvey. Satellite Network Performance Measurements Using Simulated Multi-User Internet Traffic. In *Proceedings of the Seventh International Conference on Telecommunication Systems*, March 1999.

[KOA00] Hans Kruse, Shawn Ostermann, and Mark Allman. On the Performance of TCP-based Data Transfers on a Faded Ka-Band Satellite Link. In *Proceedings of the 6th Ka-Band Utilization Conference*, June 2000.

[McL00] Neal McLean. Geo-Stationery Orbits. Technical report, Society for Broadcast Engineers Chapter 24, Inc., 2000.

[NAS] NASA ACTS. NASA's Advanced Communications Technology Satellite. http://acts.grc.nasa.gov/.

[Ost97] Shawn Ostermann. *tcptrace*, 1997. Available from http://www.tcptrace.org/.

[Pax96] Vern Paxson. tcpslice, 1996. ftp://ftp.ee.lbl.gov/tcpslice.tar.gz.