

Distributed Testing and Measurement across the Atlantic Packet Satellite Network (SATNET)

Karen Seo, Jon Crowcroft, Pål Spilling,
John Laws and John Leddy

June 6, 1988

ABSTRACT – The analysis of the test and measurement of TCP/IP performance over the Atlantic Packet Satellite Network (SATNET) is described. Both the methodology and tools as well as the results and their analysis are discussed. Because of the Internetwork nature of the environment, the tests were designed to allow the SATNET Measurement Taskforce to look at the effects of each component of the end-to-end path, e.g., local networks, gateways, SATNET, and protocol layers. Results are given for the IP service provided by SATNET as a function of offered load and for TCP behavior as a function of offered load and the underlying IP service.

KEYWORDS – measurement, testing, IP, TCP, Atlantic Packet Satellite network (SATNET), performance, throughput, delay, packet loss

1 Introduction

The Atlantic Packet Satellite Network (SATNET) [12,17] is an Internet [16] component that connects networks in Europe and the United States by two fully-interconnected satellite channels. Although SATNET has been operational for many years, opinions have varied on how well the network performs and on the effect of SATNET on Internet communication. In particular, questions have been raised as to the source of delay experienced by people using TELNET and FTP over SATNET. There have been several studies of SATNET performance in previous years [2,9,18]. However, since these earlier measurements were taken, SATNET has undergone a number of modifications, e.g., the packet switches, gateways, and satellite transponder have changed. As a result, in early 1985, the SATNET Measurement Taskforce was formed to study the problem and isolate its causes. The members have included people from the University

This research was partially supported by DARPA under Contract MDA903-83-C-0131. This paper has been approved for public release and unlimited distribution.

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.

College London, U.K. (UCL), the Royal Signals and Radar Establishment, U.K. (RSRE), the Norwegian Telecommunications Administration Research Establishment, Norway (NTA-RE), the Italian National Research Council, Italy (CNUCE), the German Aerospace Research Establishment, West Germany (DFVLR), and Bolt Beranek and Newman, U.S. (BBN). This taskforce has examined the end-to-end behavior seen by users by characterizing the IP-level service provided by SATNET and other components of the end-to-end path, measuring the corresponding TCP performance obtainable given this IP service, and analyzing the interaction between the two layers.

As part of our investigation, we isolated and corrected several problems in SATNET relating to throughput and packet loss. This work is described in a final report to the SATNET community [15]. However, in this paper, we focus on the aspects of our work that we feel have relevance to the Internet in general. We begin by describing the objectives of the taskforce, the SATNET environment, and the methodology and tools used to separate out the effects of the different network components. We then present the results of the taskforce's measurements. The tests centered on the protocol layers (IP and TCP) that we felt provided the clearest picture of the network performance.

2 Objectives of the Measurements

The primary aim of the taskforce was to examine user-level performance over SATNET. However, we wanted to do these measurements in a way that would be independent of the application-level process (e.g., TELNET or FTP) and would provide a more general understanding of the behavior of the underlying network.

To accomplish this, we chose to focus on measurements at lower protocol layers. TCP/IP is the common element under most applications in the DARPA Internet. So, we decided to concentrate on first characterizing SATNET's IP-level performance (delays, throughput, and packet loss) under different offered loads, and then on analyzing how the TCP parameters and algorithms interacted with this IP service. This approach allowed us to avoid being biased by the details of particular application protocols, but still look at end-to-end performance and see the effects of the network path. It also meant that the results would be relevant on a more general basis in other network settings.

Accordingly, we decided on the following goals for our measurement plan.

- Characterize the IP performance of SATNET itself
- Explain, and where possible, improve the IP behavior of SATNET

- Characterize the IP performance of the Internet path followed by a packet going between a source host and a destination host on different networks connected by SATNET
- Characterize the TCP performance of a packet going between a source host and a destination host on different networks connected by SATNET
- Explain, and where possible, improve TCP performance. In particular, examine the relationship between TCP performance and the underlying IP service in order to gain insight into the factors limiting end-to-end performance.

3 The SATNET Environment

SATNET is an interesting testbed for measuring performance because it differs from other long haul networks. Data is broadcast over a shared satellite channel rather than sent from one packet-switching node to another via land lines. Thus, delay and error characteristics are different. This section gives an optional overview of SATNET technology. It provides background on technical points that are relevant to understanding the measurements described later in the paper. See [10,11,12] for additional detail on any of these topics.

3.1 SATNET topology

SATNET is a trans-Atlantic, packet switching network with four nodes, each consisting of a Satellite Interface Message Processor (SIMP) and a modem/codec unit [12,17]. The nodes are fully interconnected via two multi-access 64 Kbits/sec Intelsat V channels. These channels are managed independently, and user traffic is assigned to the channel that has less queued traffic. SATNET is connected to the ARPANET, and to networks at the user sites, via Butterfly Internet Gateways. The result is essentially a backbone network over the ocean interconnecting a set of local networks.

The SATNET topology can be controlled from a remote host. Connectivity between SIMPs, and between SIMPs and attached gateways can be changed by looping a SIMP's satellite channel or gateway interfaces. This control was useful when we ran tests with only two SIMPs on the air or when we needed to throttle background traffic from attached local networks.

3.2 How the satellite channel access algorithm affects performance

The way the SIMP schedules transmissions determines the kind of delays the packet may encounter as a function of traffic load, packet arrival time, etc.

3.2.1 Priority Oriented Demand Assignment (PODA)

Bandwidth on SATNET channels is allocated to each of the four SIMPs using the Priority Oriented Demand Assignment (PODA) scheduling algorithm. PODA makes efficient use of a channel by rapidly adapting the allocation of bandwidth to the SIMPs. It assumes that the load they offer will be highly dynamic and unpredictable. PODA also provides robustness by distributing the scheduling.

To achieve these goals, PODA requires a SIMP to broadcast a "reservation" for channel space to all the SIMPs in SATNET for each packet accepted from an attached host. The reservation describes the packet's size, priority, and transmitting SIMP. Assuming no downlink errors, the SIMPs all execute the same ordering algorithm on an identical sequence of reservations, and reach the same packet transmission schedule. When a packet's transmission time arrives, the SIMP that transmitted the reservation transmits the packet; all other SIMPs remain quiet for the duration of the packet transmission.

Time on each satellite channel is divided into "PODA frames" of approximately .3 seconds, each containing two subframes. The "data subframe" represents the channel bandwidth that is scheduled by the PODA algorithm; during this time, data packets are transmitted. A data subframe can contain up to 20 minimum-sized packets or up to 5 maximum-sized packets. The "reservation subframe" contains a time slot for each SIMP during which it can send up to two reservations. In addition, two reservations can also be "piggybacked" on each data packet.

3.2.2 Propagation delay to the satellite

One of the principal factors affecting delay and throughput is the approximately .26 second propagation delay to the satellite. This is much greater than the delay of terrestrial lines. Since each packet transmission requires a reservation transmission, the minimum SATNET end-to-end propagation delay is about .52 seconds. Queuing delays imposed by processing add a further .08 seconds. Thus packets experience a minimum delay of .6 seconds. Additional factors can add an average of about .2 seconds for an average delay of about .8 seconds.

3.2.3 Available bandwidth

Only part of the channel bandwidth is available for data transfer. Reservation subframes, inter-packet time padding and "burst preamble" required by the modems, the SIMP to SIMP protocol, and the checksum leave only about 37 Kbits/sec out of 64 Kbits/sec channel. This gives a total of 74 Kbits/sec available for gateway to gateway communication.

3.2.4 Bit error rate

In SATNET, in addition to any packet loss that might occur because of congestion, there is inherent packet loss from bit errors. This error rate is a function of the up- and downlink signal-to-noise ratio, the adjustment of the modems, and the forward error correction code in use. The resulting bit error rate is 10^{-6} , which produces a packet drop rate of approximately .3% for maximum size packets. Smaller packets have a lower drop rate since they have fewer bits and hence a lower probability of error per packet.

4 Methodology

A major accomplishment of our work is the experimental approach and the measurement tools we developed. In order to reach our goals, we had to solve a variety of problems that apply to any tests done in a real network setting, and especially in a distributed measurement program. In particular, we had to design tests that were scientifically sound (e.g., controls, known values for independent variables such as background traffic and topology) and whose results were reproducible and applicable to the Internet.

Given the goal of explaining delays seen by users and measuring SATNET performance characteristics, the taskforce first had to decide what parameters to measure and how to measure them. We agreed to use delay, throughput, packet loss, and availability as performance measures. We then used results from earlier tests and some anecdotal evidence to decide where and how to make measurements. This helped us locate bottlenecks and design the measurement plan. Our measurement plan involved iterating on the following steps:

- create a hypothesis for what should be happening along a particular part of the network path and at a particular protocol level
- perform measurements
- identify problems in the network path
- fix the problems identified by the tests

- repeat measurements to verify that the problems were fixed

The following sections provide more detail on the measurement plan and tools that were used.

4.1 Measurement Plan

Because of the distributed nature of the network and user sites, we found it imperative to carefully coordinate and plan the measurement work. Preliminary measurements and discussions were used to identify the following critical aspects of the test plan. [For additional detail, see [3,15,20].]

- which independent variables to record and what values to use – average packet length and distribution of lengths, average packet rate and distribution of interarrival times, bit error rate, time of day, background traffic, network topology, and number of satellite channels
- what data path to use – this included all source/destination combinations of local hosts, SIMPs, and gateways
- which performance parameters to measure – round-trip delay, throughput, packet loss, and host-to-host availability
- which network layers to examine – IP and TCP
- what IP and TCP tools to use – monitoring tools, echoers, and traffic generators
- what experimental procedure to follow, e.g., use of benchmark tests and controls to ensure comparability of results across experiments.

In general, we looked at delay, throughput, and packet loss. Measurements were done for a range of packet rates and for a range of packet sizes going from the minimum IP packet size to the maximum SATNET packet size. These were typically done by generating traffic at the source and sending it to an echoer at the destination. Some of the key paths used are shown in Table 1. The hosts were typically on local area networks connected to SATNET by Internet gateways (GWY).

| | Prot. | Source | Destinations |
|----|-------|---------|--|
| 1. | IP | SIMP | satellite echo, remote SIMP |
| 2. | IP | Gateway | local SIMP, remote SIMP, remote GWY |
| 3. | IP | Host | local GWY, local SIMP, remote SIMP, remote GWY |
| 4. | TCP | Host | local SIMP, remote SIMP, remote Host |

Table 1: Key paths

By comparing results from these tests, we were able to isolate the effects on delay and throughput from packet size, offered load, and each part of the network path seen by a packet traveling from one host to another. Also, by comparing results from tests 1, 2, and 3 with those from 4, we were able to examine the interactions of TCP with the underlying IP service. We were also able to look at the effects of packet loss, time of day, and background traffic.

As mentioned previously, the taskforce then performed independent and coordinated measurements to characterize SATNET performance. This was an iterative process with each round of results helping to define the next round of tests. In particular, it was necessary to obtain benchmarks for use in selecting values for independent variables such as background traffic, time of day, and rate of packet generation. Where possible, all tests were done using paths involving only SIMPs, gateways attached directly

to SATNET and hosts on networks connected to SATNET. Although the problems encountered by the SATNET community were for traffic over more complicated paths, e.g., ethernet to SATNET to ARPANET to MILNET to ethernet, time and resource limitations constrained us to looking at simpler topologies.

We also took advantage of a testbed at BBN which allowed us to do experiments in a controlled environment. This testbed consisted of two SIMPs, an Internet gateway and a channel simulator connecting the SIMPs. Overall, we stressed reproducibility to make sure results were valid. In addition to comparing results obtained in the testbed with those obtained in the real SATNET, we also compared the results obtained with different tools.

4.2 Coordination of test and measurement

One of the obstacles that the taskforce encountered was having to compare and evaluate the results produced by disparate tools and experimenters. We mention this for the benefit of other experimenters working in similar situations. Most tests reported in the SATNET literature involved one or two people working at the same site who therefore could easily avoid problems of running non-comparable tests. Although it may seem obvious, we learned first hand that it was helpful to have a fixed, agreed upon format for reporting all experiments. In particular, in order to evaluate or compare experiments it was essential not to omit any information on the experimental setting – network configuration, background traffic, parameters of the offered load, time of day, etc. Having a pre-defined experimental format encouraged taskforce members to use controls to ensure that tests performed on one day were comparable to tests done on another day and that tests used adequately large sample sizes. For example, in some cases, a benchmark test was done at the beginning of the day's test runs. In other cases, periodic benchmark tests were run throughout the time the measurements were being performed.

4.3 Tools

A variety of tools were needed to do the tests and analyze the results [20]. In addition to tools for measuring performance parameters, tools were required that could give a picture of exactly what was happening to packets that was leading to the observed performance, e.g., retransmissions. The following paragraphs provide a brief overview of the tools we used. Specific tools are described in more detail in some of the later results sections.

4.3.1 Existing tools

Each SIMP implements echo hosts that return packets to the source and message generators and sinks that can be used to compile statistics. The SIMPs also collect and report status and information on error rates, traffic loads and channel quality.

The Internet gateways (LSI-11, PDP-11, Butterfly) provide ICMP/IP traffic generators.

Van Jacobson provided a tool for observing packets on a TCP connection, which allowed us to observe a TCP's behavior without altering the actual implementation. This program runs on a machine on the same LAN as either the source or sink of a connection. It filters all the packets for a given connection for later analysis of the timing of packet transmissions, acknowledgments, retransmissions and so on.

4.3.2 Tools developed by the taskforce

In order to be able to generate controlled traffic, including high enough traffic loads to stress SATNET, UCL developed a program which attempts to drive random data from memory as quickly as possible over a TCP connection. The source and sink machines were originally LSI/11s. Later they were high speed workstations easily capable of reaching the limitations of the 64 Kbits/sec SATNET.

UCL developed an instrumented version of TCP which allowed a traffic generator to set parameters such as maximum segment size (MSS), maximum window sizes, retransmission timeout estimators, and so on. This ran on LSI/11's under a local real-time operating system. The traffic generator was integrated with the TCP. [See section 7 for more explanation of TCP parameters, also see [5].]

UCL similarly instrumented the 4.3BSD TCP by providing a control path for a user to access all the state variables for any given connection, and access them on a network or timer event basis.

NTA-RE/NCC developed an ICMP/IP tool that runs on a Vax or SUN [10]. The measurement tool consists of three processes – an interactive, command-driven interface for setting up experimental parameters; a probe process that measures round-trip delay at a low rate and which is used to detect changes in the network environment and to facilitate comparisons between tests; and a traffic generator/sink/logger process that generates ICMP packets of a specified type (echo request or echo reply) and size and transmits the packets to a list of specified destinations. (See section 6.2 for more detail.)

5 IP Performance – SATNET

The first step in our measurement plan was to measure and characterize the IP level service provided by SATNET itself. This included the SIMP backbone and attached gateways.

5.1 Measurement plan

The performance characteristics of the SIMP to SIMP link and the Gateway to SIMP link were measured in order to understand their contribution to the end-to-end performance experienced by hosts. The important parameters were throughput, delay, and packet loss. We made measurements for each of these parameters for the SIMP to SIMP links, but the Gateway tools only permitted measurements of throughput and packet loss for Gateway to SIMP links.

5.2 Subnet Measurement Tools

The tools [2,9] that were used were already present within the SIMPs and Gateways. Some modifications were made to them to gain more information for particular tests. The tools included message generators, sinks, and echo hosts in the SIMPs; and message generators and sinks in the gateways (all IP-datagram based). Each SIMP has both a local echo host and a satellite echo host. The local echo host simply returns the packet. The satellite echo host first sends the datagram up on the channel to the satellite and then returns it. Monitoring information from the SIMPs and gateways is reported to a Network Monitoring Host located at BBN. The monitoring reports included traffic information, link status, exception reports, and CPU utilization. These reports were used as a source of data and as a check for abnormal behavior of the SIMPs during tests.

5.3 Measurement Results

The following sections describe SATNET's IP characteristics that bound the performance of higher level protocols, e.g., TCP. This IP performance also defines the environment which higher level protocols have to handle, e.g., bit error rate.

Short Term, Peak Throughput (SIMP to SIMP link) – To determine how closely the SIMP approached the theoretical maximum throughput, we measured the maximum number of packets that could be sent during a monitoring period irrespective of delay and packet loss. We ran tests from the Roaring Creek SIMP to the satellite echoer using the SIMP message generator, which was turned on full, offering 1 packet per virtual slot or 1 packet every 10.25 msec. The test was run for 3 packet sizes;

52 bytes, 148 bytes, and 256 bytes (IP packet lengths including IP header). Monitoring reports gathered from the SIMP every 84 seconds reported the number of packets actually sent over each channel. There were no monitoring reports generated from the other SIMPs, thus no interference between measurement traffic and monitoring traffic on the satellite channel. From these reports the actual channel throughput was calculated.

At the largest packet size (256 IP bytes), measured throughput comes very close to the theoretical throughput. For 148 byte packets we are within 20% of theory. At 52 bytes, however, monitoring reports of CPU utilization showed that the SIMP's throughput was CPU limited. It is expected that throughput is packet limited and will deviate increasingly from the theoretical maximum as packet size decreases. The measured peak rates are shown in Table 2.

| IP (bytes /pkt) | Theoretical Maximum (Kbits/s) | Measured Peak Rate (Kbits/s) | Measured Steady Rate (Kbits/s) | "Knee" as % of Peak |
|-----------------|-------------------------------|------------------------------|--------------------------------|---------------------|
| 52 | 36 | 26 (74%) | 21 (59%) | 79+% |
| 148 | 64 | 53 (82%) | 47 (74%) | 90+% |
| 256 | 75 | 71 (95%) | 59 (78%) | 83+% |

Table 2: SIMP to SIMP Throughput

Long Term, Steady State Throughput (SIMP to SIMP link) – To determine the maximum usable throughput of the SIMP, measurements were made to find the maximum throughput where less than 1% of the packets were dropped. A problem with these tests was the granularity of the SIMP message generator. The steady state maximum throughput was found by measuring the highest load the SIMP message generator could offer that did not exceed the maximum measured peak throughput. The usable throughput of the SIMP is therefore between our measured values and the values for maximum peak throughput. The throughput for the 1% drop rates are therefore artificially low. Table 2 shows the lower limit of the usable throughput. The usable percentage of theoretical throughput decreases with packet size as the throughput becomes packet limited.

Delay vs. Offered Load Curve (SIMP to SIMP link) – The delay encountered by packets as offered load increases gives a good picture of how a network degrades with increased load. To determine the shape of the graph for the SIMP-to-SIMP link, we measured delay from the Roaring Creek SIMP to the Goonhilly SIMP echoer for three packet sizes. The minimum, maximum, and average delays were graphed for offered loads above, below, and near the maximum channel throughput. The resulting graphs [15] showed the knee of the curves occurring very close to the maximum measured throughput of the channel. Average delay decreased as throughput increased until the knee was reached, as expected of the PODA access method [2]. Table 2 shows the maximum throughput point as a percentage of peak throughput.

Throughput (Gateway to local SIMP link) – To find the throughput for the link from a gateway to SATNET, we measured the maximum number of packets received and sent between a gateway and a local SIMP echoer. We used the message generator in the CSS gateway to send traffic to the local echo host in the Roaring Creek SIMP. The message generator was turned on full (its output queue was kept full) and the number of packets transmitted was measured for an 84 second period.

We investigated the cause of the throughput limit shown below, but did not have time to complete this effort. This limit is not due to the inability of the SIMP to process higher packet rates. A test using two gateways sending traffic through the same SIMP measured throughput rates approximately double those using one gateway. It was also found that for different line speeds the percentage of usable bandwidth for a particular packet size remained approximately constant.

| IP (bytes /pkt) | Raw Line Speed (Kbits/s) | Measured Gwy↔SIMP (Kbits/s) | Measured Gwy↔Remote (Kbits/s) |
|-----------------------|--------------------------------|-----------------------------------|-------------------------------------|
| 52 | 50 | 16 (32%) | 15 |
| 125 | 50 | 27 (54%) | 26 |
| 246 | 50 | 41 (82%) | 39 |

Throughput (Gateway to local SIMP to remote SIMP)

When a satellite hop was included in the gateway to SIMP throughput measurements, there was no appreciable change in the gateway to SIMP throughput. The limiting throughput is therefore the gateway to SIMP link with the maximum available bandwidth given in the table above.

Delay distribution (SIMP to SIMP link) - The PODA channel access method used in SATNET [17] yields relatively large variations in the network delay. This is caused by the variable time a packet has to wait both for a reservation opportunity and, after its reservation has been received, for a data transmission opportunity. At low packet rates (reservations are not piggybacked onto data packets), the variability is primarily a function of the arrival time of a packet relative to the next reservation opportunity. The PODA protocol provides an opportunity for each site to send a reservation with at most about .38 sec delay, with a mean delay of about .16 sec [15]. At higher packet rates, piggybacking decreases the average delay for a reservation opportunity. Average delay therefore decreases with increased load until queuing delays increase.

The propagation time from an earth station to the satellite and back (hop) is about .26 sec. Since PODA requires two hops for a packet to be successfully transmitted and received, the propagation time adds .52 seconds of delay. Tests run in the SIMP show an internal processing time of .08 sec per packet. The transmission time to send a packet out on the channel ranges from 0 to .03 sec depending on packet length. At light loads, the wait to send ranges from 0 to .09 sec. Therefore, the minimum possible transmission time is $(.52 + .08) = .6$ sec. If you add the average time to wait for a reservation opportunity, .16 sec, plus the average time to wait to send, .045 sec, plus the average time to clock a 52 byte packet onto the channel, .006 sec, you get an average transmission time of about $(.6 + .16 + .045 + .006) = .81$ sec for a 52 byte packet.

Measurements of actual delay for a 52 byte packet from the Roaring Creek SIMP to the Satellite echoer found a minimum of .60 sec and a maximum of 1.67 seconds. The average RTT was .82 seconds, which is in good agreement with the prediction above. Out of 2,633 packets, only two packets experienced anomalous delays, one at 1.2 seconds and one at 1.67 seconds. The delay distribution did not vary significantly with packet size.

Packet loss - The bit error rate on the channels is a function of the up and downlink signal to noise ratio, the adjustment of the modems, and the forward error correction. The resulting predicted bit error rate is 10^{-6} , which produces a packet drop rate of approximately .3% for maximum size packets. The actual measured packet drop rates ranged from 0 to .4% for a 600 packet sample. The measured gateway to SIMP packet drop rates due to noise are negligible.

| IP Packet Size (bytes) | Pkt Generation Rate (pkts/sec) | Packet Loss (%) |
|---------------------------|-----------------------------------|-----------------|
| 56 | 1 | 0.0 |
| 72 | 1 | 0.1 |
| 128 | 1 | 0.15 |
| 192 | 1 | 0.35 |
| 256 | 1 | 0.4 |

6 IP Performance - Internet

In addition to characterizing the IP performance of the SATNET subnet, we measured the IP performance over the end-to-end path. This is the path followed by user traffic going from hosts on local area networks interconnected by SATNET.

6.1 Measurement Plan

The IP datagram is the basic vehicle for the transportation of information in the Internet between network access points. Thus the behavior of this network-layer transport mechanism defines the baseline performance upon which the end-to-end transport layer and applications are built. It is therefore of importance to characterize this performance and understand its behavior, in order to interpret the results of higher-level measurements.

The most important parameters characterizing the behavior of the IP transport system are network throughput and delay, packet loss rate, and the degree of packet sequence disorder as packets traverse the network. All these should be measured as function of packet size and traffic load in the network.

6.2 IP Measurement Tool

An IP measurement package [10] was developed to run under Berkeley Unix on a VAX or a Sun workstation. It makes use of the ICMP echo.request/echo.reply facility to have packets looped back from specified remote hosts. This mechanism is part of the IP/ICMP protocol implemented in all hosts and gateways in the Internet system. In addition, echo hosts are available in the SIMPs for general looping of IP traffic.

The measurement facility consists of three parts: an interactive, command-driven interface for setting up experimental parameters and starting the experiments; a traffic generator and traffic sink; and a logger process. This facility can generate echo.request and echo.reply packets and send them to one or a specified set of destination hosts. The echo.reply packets are used when the packets are sent to the echo hosts built into the SIMPs. Packet size and frequency are selectable with either constant or random values with specified means. The destination host, packet size, packet frequency, TCP transmit sequence number and time of departure are recorded in an experiment file. The traffic sink absorbs the returning packets and records their sequence number and time of arrival in the experiment file. At the end of each experimental run a summary of the run is produced, but the main processing of the logged data is performed off-line. The traffic generator can either be set up manually or driven from a prepared command file. There is also a probe process that generates echo.request packets at a low rate and sends them to a specified reference host. The traffic load caused by the probe process is low enough that it does not contribute measurably to the traffic load on SATNET. The calculated round-trip time for the probe traffic can be used as a reference to verify that the network condition remained reasonably constant during the measuring period. Different sets of measurements can also be compared to each other more reliably.

6.3 Network Configuration

Figure 1 shows the network configuration relevant to the IP round-trip time measurements. The measurement host is a VAX-750 located at one of the Ethernets at NTA-RE. This network is interconnected with another Ethernet which in turn is connected via the NTA-RE Butterfly Gateway and a 48 Kbits/sec line to the SIMP at the Tanum ground station. This is a typical situation for hosts and workstations at NTA-RE.

6.4 Measurement Results

Throughput - The end-to-end throughput over SATNET was determined from the measured round trip time (RTT) as a func-

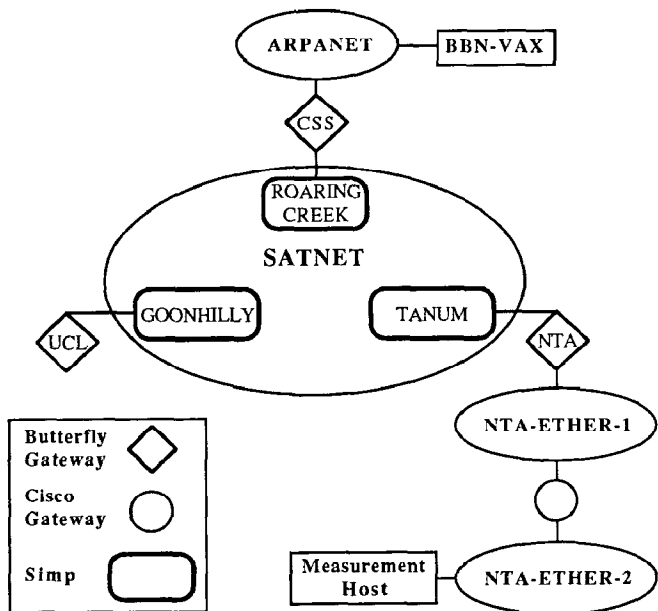


Figure 1: Relevant Network Configuration

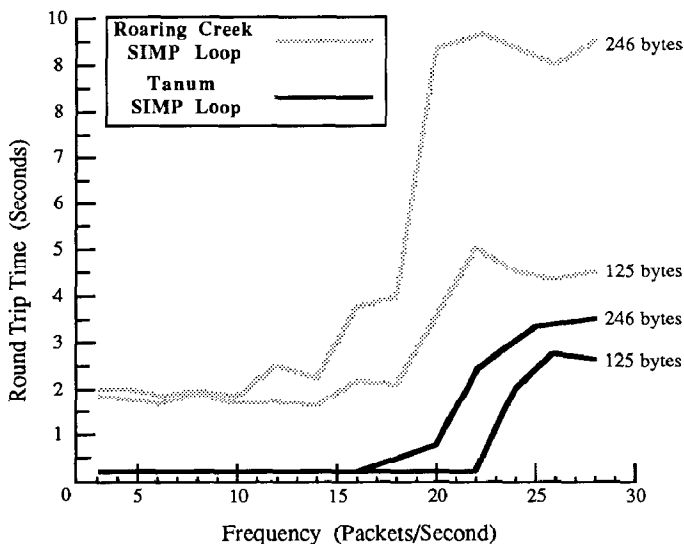


Figure 2: Throughput Measurements

tion of traffic load for two packet sizes, 125 bytes and 246 bytes. As shown in Figure 2, the sharp increase in RTT indicates the approximate throughput limit for each packet size. As a reference, the throughput for the 48 Kbit/sec line from NTA-RE to the Tanum SIMP was measured to verify that it did not represent a bottleneck. For a packet size of 246 bytes (close to the SATNET maximum packet size of 256 bytes), the throughput from NTA-RE to the Tanum SIMP and looped back is about 39 Kbits/sec or about 81% of the line capacity. This is consistent with the gateway to SIMP link throughput reported in section 5.3, and shows that for large packets this link does not limit the available throughput across SATNET. The capacity across SATNET is represented by the knee in the RTT curve of packets that were sent from NTA-RE through the Tanum SIMP, across SATNET to the Roaring Creek SIMP, and looped back. It should be noted that this traffic is traversing the shared channels twice. For 246 byte packets, this knee occurs somewhere between 60 Kbits/sec and 70 Kbits/sec on the channel. This is consistent with the long term steady state throughput and short term peak throughput reported in the previous chapter. For 125 byte packets, the

knee occurs between 35 Kbits/s and 40 Kbits/s, or about 50% to 55% of theoretical expectation. This appears to be caused by a packet/sec limit between the gateway and the SIMP (see Gateway to local SIMP Throughput in section 5.3).

Distributions of Round trip times - The average measured round trip times for two packet sizes (125 and 246 bytes) and three packet frequencies (1, 5 and 10 pkt/sec) are shown in Table 3. Packets were looped off the local Butterfly Gateway (BGWY), the Tanum SIMP, the satellite, the Roaring Creek SIMP, the DCEC Butterfly Gateway (DCEC BGWY), and BBN-VAX.

| Echo location | Packet size | Packets/second | | |
|--------------------|-------------|----------------|------|------|
| | | 1 | 5 | 10 |
| BGWY | 125 | 43 | 43 | 44 |
| | 246 | 47 | 48 | 47 |
| Tanum SIMP | 125 | 146 | 146 | 156 |
| | 246 | 193 | 197 | 203 |
| SATNET Echo | 125 | 1021 | 984 | 970 |
| | 246 | 1090 | 1059 | 1053 |
| Roaring Creek SIMP | 125 | 1868 | 1758 | 1717 |
| | 246 | 2003 | 1873 | 1865 |
| DCEC BGWY | 125 | 2016 | 1963 | 2062 |
| | 246 | 2166 | 2169 | 2305 |
| BBN-VAX | 125 | 3003 | 2985 | 2999 |
| | 246 | 3166 | 3177 | 3504 |

Table 3: Average measured round-trip times (msec)

If we use the satellite echo RTT prediction from the previous chapter of about .80 sec and add the transmission time for the gateway-SIMP line (48 Kbits/sec) of 2 x .04 sec, we expect an average RTT of about .88 sec for the 246 byte packets. This is about .17 sec smaller than the measured value of 1.05 sec. This is in part because of the forwarding delays in the gateways in the path between the measurement host and SATNET, and in part because of the processing time involved with the access protocol used between the SIMP and the gateway.

A similar estimate for the round trip time looping off the Roaring Creek SIMP gives 1.68 sec. The measured value is 1.87 sec. Thus the estimated and measured values are in good agreement with each other. Looping off the DCEC BGWY should add the packet transmission time both ways on the 50 Kbits/sec line from Roaring Creek to DCEC, approximately .08 sec, plus the loop processing time in the BGWY, approximately .02 sec. The expected RTT is therefore 1.78 sec, compared to the measured value of 2.1 sec.

The RTT measured looping off BBN-VAX is included to show a typical response from a host on ARPANET near one of the interconnection points between ARPANET and SATNET. RTT to other locations varies substantially, depending on the instantaneous traffic load and the locations in ARPANET. RTT can be as high as 20 - 25 sec in extreme situations.

Figure 3 shows typical measured RTT distributions, for moderate traffic loads, as one progresses further and further away from the measurement host. As expected, the distribution broadens and the tail of packets experiencing higher delay increases as one moves away from the measurement host. This tail is difficult to explain, but is in part due to queueing delays at the transmission side of SATNET. It cannot be explained by loss of reservation packets because of noise. The observed loss rate of data packets is very small, and the effect of noise on the reservation packets should be substantially smaller. A few packets out of a thousand traversing SATNET experience a delay greater than 5 to 10 sec.

Disordering of packet sequence - Due to SATNET's two channel operation, the sequence of packets traversing SATNET can be permuted. The loss of reservations due to channel noise is considered to be too small to contribute significantly. The

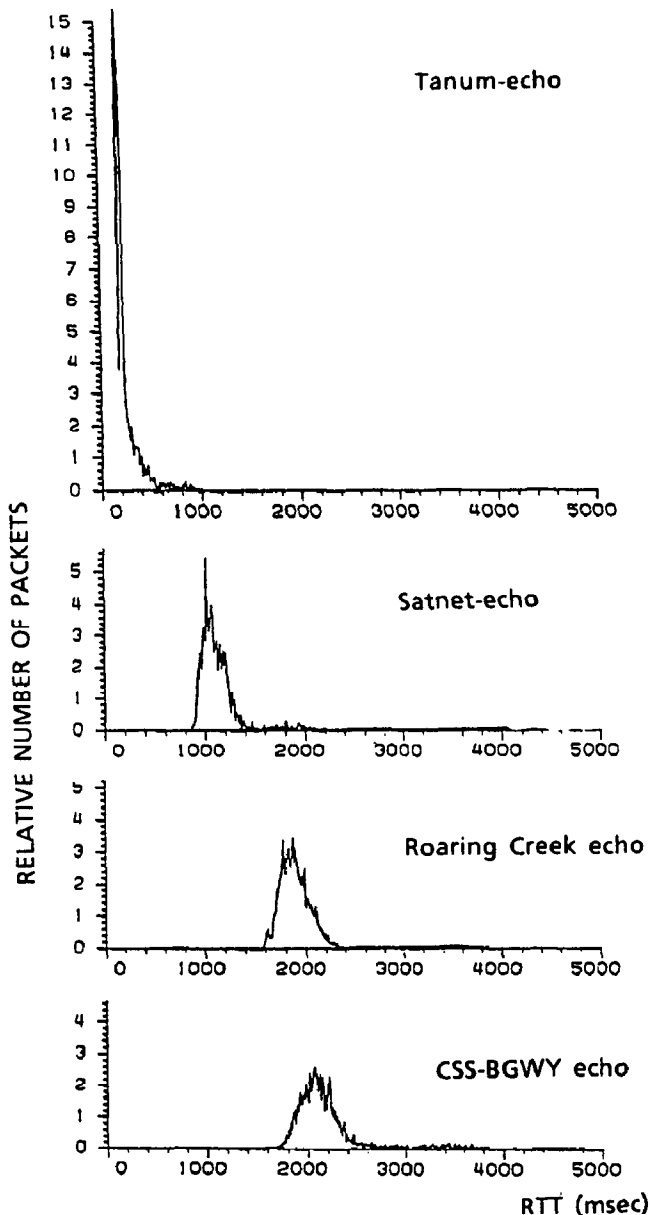


Figure 3: Typical round-trip time distributions

probability of permutation increases with load. Table 4 shows the observed permutations for different packet sizes, frequencies, and paths. As expected, the degree of disorder increases with network distance from the measurement host.

| pkt frequency | 125 bytes | | | 246 bytes | | | # perm | # pkt |
|---------------|-----------|-----|-----|-----------|----|-----|--------|-------|
| | 1 | 5 | 10 | 1 | 5 | 10 | | |
| SATNET Echo | 0 | 62 | 185 | 0 | 22 | 139 | 408 | 3400 |
| Roaring Creek | 0 | 94 | 204 | 0 | 80 | 222 | 600 | 3400 |
| CSS-BGWY | 0 | 110 | 239 | 0 | 75 | 241 | 665 | 3400 |
| BBN-VAX | 0 | 63 | 267 | 0 | 84 | 266 | 680 | 3400 |

Table 4: Observed packet sequence permutations

For packets traversing SATNET twice, each traversal should contribute equally to permutations. For packets going through SATNET once (looping off the satellite) we observed that 120 packets out of 1000 packets had been permuted. For packets traversing SATNET twice (looping off Roaring Creek) 174 out of 1000 packets had been permuted. Assuming that the probability for permutations adds quadratically, the ratio between the two

observed permutations should be equal to $\sqrt{2}$.

Observed ratio = $600/408 = 1.47$, expected ratio = 1.41

Packet loss – At low to moderate traffic loads, the main contributor to loss of packets is channel noise. At higher loads, the main contributor is the limited number of buffers available at the transmission side of the SIMPs and the decision to drop packets rather than use some other strategy for handling the overload. Table 5 shows the observed losses of packets for two packet sizes at moderate traffic loads. As expected, the loss rate increases with packet size. Using this data we calculate a bit error rate of $.7 * 10^{-6}$, which is in good agreement with other observations.

| Loop location | Packet size | Obs. loss | Total pkts | Loss per 1000 pkts |
|----------------|-------------|-----------|------------|--------------------|
| Satellite echo | 125 | 3 | 3400 | 0.9 |
| | 246 | 3 | 3400 | 0.9 |
| Roaring Creek | 125 | 5 | 3400 | 1.5 |
| | 246 | 10 | 3400 | 2.9 |
| CSS-BGWY | 125 | 23 | 3400 | 6.8 |
| | 246 | 36 | 3400 | 10.6 |
| BBN-VAX | 125 | 14 | 3400 | 4.1 |
| | 246 | 24 | 3400 | 7.1 |

Table 5: Observed Packet Loss Across SATNET

7 TCP Performance

Many applications in the Internet that require reliable communications are built on top of TCP. This makes it important to determine TCP performance in the high-delay environment imposed by SATNET. TCP has adaptive timeouts and windows, and therefore has complex behavior when the quality of the underlying internetwork service varies. It is also functionally very similar to the ISO Transport Protocol Class 4. So the TCP performance results should be of general interest.

7.1 Characterizing Performance

There are two important views of end-to-end protocol performance. End users are interested in high throughput for bulk data transfer, and low round trip delays for interactive use. SATNET has reasonable underlying throughput, but a high basic delay. Network providers are interested in high utilization of the net, together with fair sharing of network resources. SATNET is intrinsically a shared resource, being a broadcast-based medium. Most studies of TCP behavior therefore concentrate on methods of maximizing throughput and minimizing delays, while minimizing unnecessary retransmissions. Some more recent studies have looked at fairness in sharing [14]. In measuring the performance of a given TCP implementation, we need to examine all of these factors.

7.2 Parameters and Algorithms

When measuring a TCP implementation, we need to systematically characterize the algorithms and parameters used by the protocol so that comparison between implementations or parameter settings is possible.

The parameters considered to be most critical for the performance of TCP are as follows:

Packet sizes – TCP negotiates a maximum segment size (MSS) on opening a connection. This is usually a simple choice between local net MSS and the standard Internet MSS of 576

bytes. The optimum MSS for SATNET may not be the same as the SATNET maximum IP packet size (256 bytes SATNET maximum packet size - 20 bytes IP header - 20 bytes TCP header - 6 bytes additional overhead = 210 bytes user data). A TCP MSS that is larger than a network's maximum IP packet size leads to a savings of one IP header size for all but the first IP fragment. However, loss of a fragment leads to loss of the whole TCP segment and subsequent retransmission of the segment. There is therefore a tradeoff between reducing packet overhead and increasing retransmissions, which depends on the SATNET packet loss rate. Another consequence of setting TCP's MSS higher than the SATNET IP MSS is that the lack of flow control between one IP fragment and the next may affect performance. We measured the performance of bulk transfers for a variety of MSS's to see if the optimum was as IP level measurements and theory would predict.

Timers - We examined two adaptive retransmission timer methods. The RSRE algorithm [21] uses the simplest possible feedback mechanism. Simple feedback from the mean measured round trip delay is used to update an estimate of the actual delay. This mechanism does not take into account the errors in the estimation method or the underlying delay due to transmission technology. Several papers [18,7,1] in the literature suggested that the mean together with the variance of the round trip time should be used as a better estimator. We compared the two algorithms.

Windows - Many early TCP implementations use the minimum of the receiver's advertised window, and the estimated "pipesize" between the transmitter and receiver. However, the receiver cannot tell the transmitter anything about network conditions, and so cannot use the same mechanism for both congestion control and end-to-end flow control. On the other hand, a transmitter can use the loss of acknowledgements as an indication of when to reduce transmissions. Unfortunately, many TCPs simply compound network congestion by retransmitting the entire window after a loss which may have been of only a single segment.

Recent developments suggest a "slow start" algorithm is useful for opening the window after loss [1]. This algorithm increases exponentially with each acknowledgement until reaching some threshold level such as the previous problem window size. Then it uses a linear increase to try and find the optimum "congestion window" without causing the window size to oscillate around the right value. The exponential to linear switch is designed to be stable in situations where the network is quick to fail and slow to recover.

We compared the throughput and number of unnecessary retransmissions for a conventional windowing TCP to those of a TCP using the slow start and congestion controlled window mechanisms.

7.3 Topology and Measurement Paths

Figure 4 shows the relevant network topology for the TCP measurement work. Tests were run for the paths listed below. The same experiments were also repeated for multiple simultaneous connections from TCP sources at UCL to sinks at one other remote site, and for differing remote sinks.

| From TCP Source at | Via IP echo at | To TCP Sink at |
|--------------------|----------------|----------------|
| UCL | local SIMP | UCL |
| UCL | "Satellite" | UCL |
| UCL | remote SIMP | UCL |
| UCL | — | CNUCE or NTA |
| CNUCE | — | UCL or NTA |

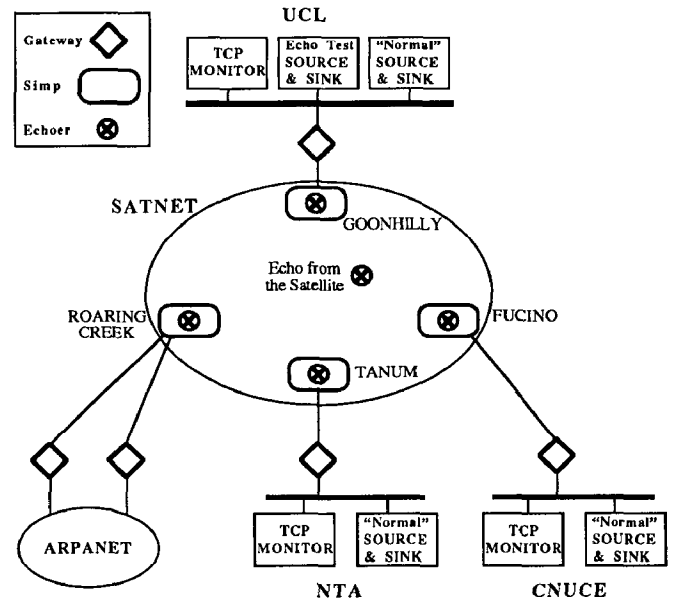


Figure 4: Traffic Paths

7.4 Results

The following sections contain the results of the tests done to measure the effect of the parameters described in the previous sections.

RTT Estimation - These experiments were done between a Sun on the UCL LAN and the satellite echo host. We used XTCP [13], (an experimental TCP based on 4.3 BSD), with an MSS of 210 octets and maximum receive window of 16 Kbytes. Figure 5 shows two RTT estimation algorithms versus actual RTT. The Time axis is half the measured or estimated round trip.

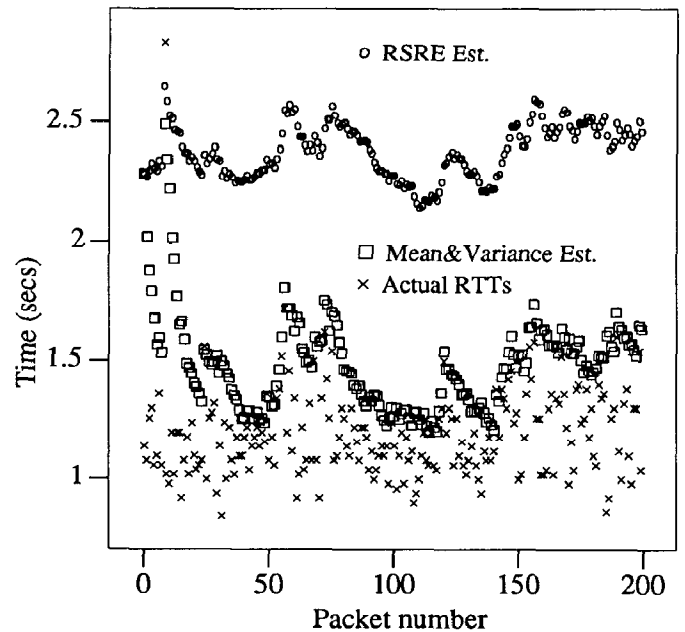


Figure 5: SRTT estimators vs. actual RTT data

With the "Smoothed Round-Trip Time" (SRTT) algorithm of RSRE, SRTT does not follow the measured RTT closely. It is higher than it should be due to the high basic delay over SATNET, which causes errors in estimation to be exaggerated.

The addition of variance to the calculation of SRTT produced a marked improvement. The SRTT much more closely follows the RTT curve. The peak near the beginning of each sequence plot corresponds with the initial high load as TCP finds a correct window (see below). In the steady state, the mean and variance of the round trip time fit closely with those seen directly at the IP level.

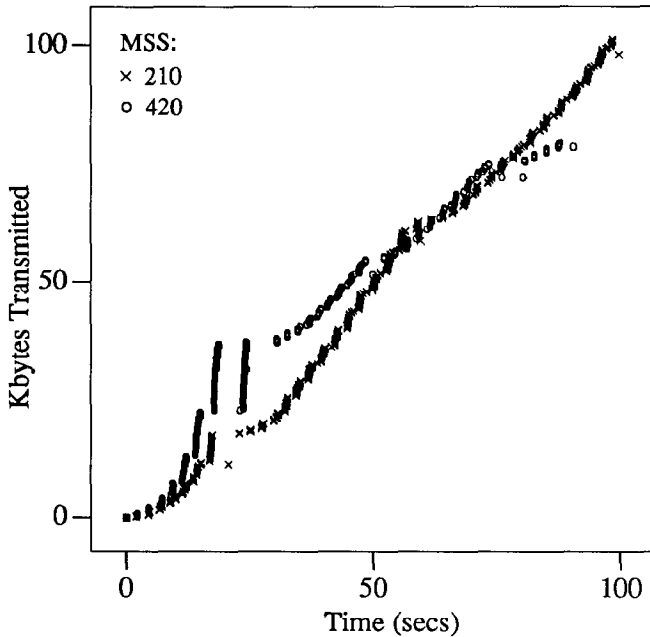


Figure 6: Effect of varying MSS on Throughput

MSS Selection - These experiments were done between a Sun on the UCL LAN and the satellite echo host. The XTCP was operated with a maximum receive window of 16 Kbytes and 2 different MSS's. Figure 6 plots Kbytes transmitted versus time for the two MSS experiments. The TCP MSS sizes chosen were one and two times the IP MSS, minus the TCP header size. In the graph, X's indicate the 1X size (210 octets), and O's indicate the 2X size (420 octets).

It was hypothesized that for the larger MSS, there might be more retransmissions due to loss of IP fragments than there would be for the smaller MSS, but that there would be more header overhead for the smaller MSS. However, the effect on throughput of MSS (at least at one times and two times the SATNET IP MSS) did not seem to be very large in these tests. One effect seen here is the more catastrophic collapse of the connection soon after startup for the case when the TCP MSS exceeds the IP MSS.

Retransmission - These experiments were done between a Sun on the NTA-RE LAN and one on the UCL LAN, using XTCP with MSS = 420 octets, and a maximum receive window of 16 Kbytes. Figure 7 shows retransmissions after loss/congestion for two retransmission strategies with Kbytes transmitted plotted against time. Dots indicate 'Standard' TCP which retransmits whole windows, and +'s indicate 'Slow Start' TCP which opens up the window slowly after packet loss.

The figure shows that for 'Standard' TCP, after a packet loss, there are a significant number of retransmissions. It also shows that for 'Slow Start' TCP there are far fewer retransmissions as the window is opened first exponentially and then linearly. Basically only the packet that was lost gets retransmitted. In SATNET, packet loss is typically due to bit errors rather than congestion, so losing one packet does not imply the loss of any of the following packets. For packet loss due to congestion, the retransmissions incurred in the first case might not turn out to be as "unnecessary".

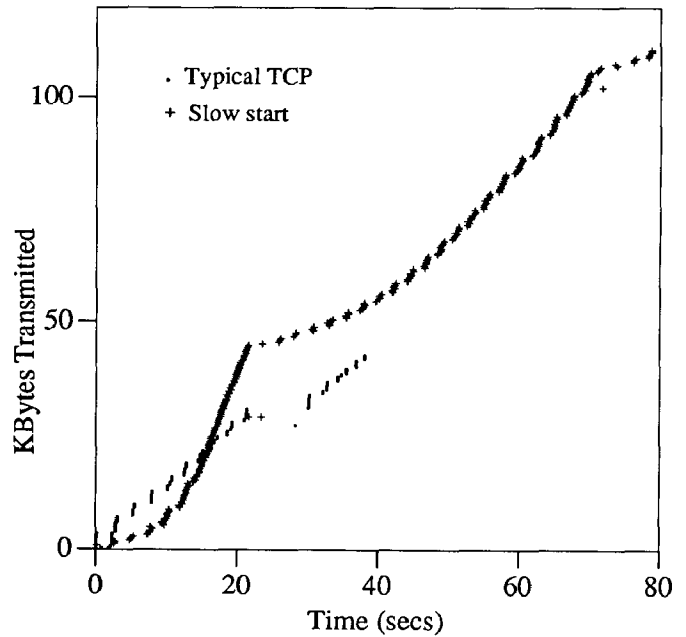


Figure 7: Starting a TCP Connection

Adaptive Window Sizing for Pipesize - These experiments were done between a Sun on the NTA-RE LAN and one on the UCL LAN, using XTCP with MSS = 420 octets, and a maximum receive window of 16 Kbytes.

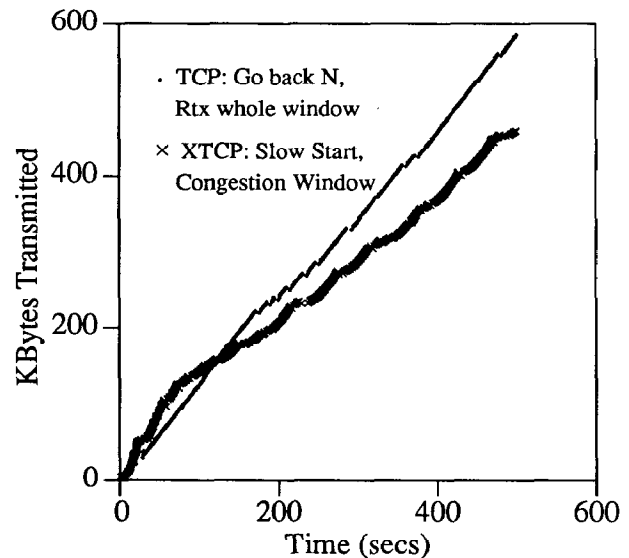


Figure 8: Longterm TCP Behavior

Figure 8 shows retransmissions after loss/congestion for two adaptive window sizing algorithms. Kbytes transmitted is plotted against time to show the differences between 'Standard' and 'Congestion Window' TCPs.

The figure (dots) shows the effects of retransmitting the whole window after a start or packet loss. If there were congestion, this strategy only makes matters worse. If there is random packet loss, as in SATNET, this causes unnecessary retransmission of packets that were not lost and lowers throughput. The figure (X's) also shows that the use of a congestion window avoids the problem of "unnecessary" retransmissions. However, when the packet loss is due to random bit errors rather than congestion, there is unnecessary throttling of transmissions and there-

fore lower throughput. Each glitch in the line is a lost packet. Counting through these we can see around a 1% packet loss rate. (Since TCP is using an MSS of two IP maximum fragments, this represents closer to 2% TCP packet loss). At this level, the exponential/linear congestion algorithm can never find the right bandwidth. Analysis shows that we can expect less than half the bandwidth that IP might achieve. These results point to the problem in interpreting packet loss as an indication of congestion in an environment where packet loss is actually caused by bit errors.

SATNET and ARPANET - This experiment was a run of XTCP over a path involving both SATNET and ARPANET, with an MSS of 512 octets, and an advertised receive window of 4 Kbytes. The path went from the UCL LAN to EDN-VAX.ARPA. Figure 9 shows the adaptation of the TCP implementation to the lower bandwidth encountered on the far side of the SATNET (possibly on entering the ARPANET.)

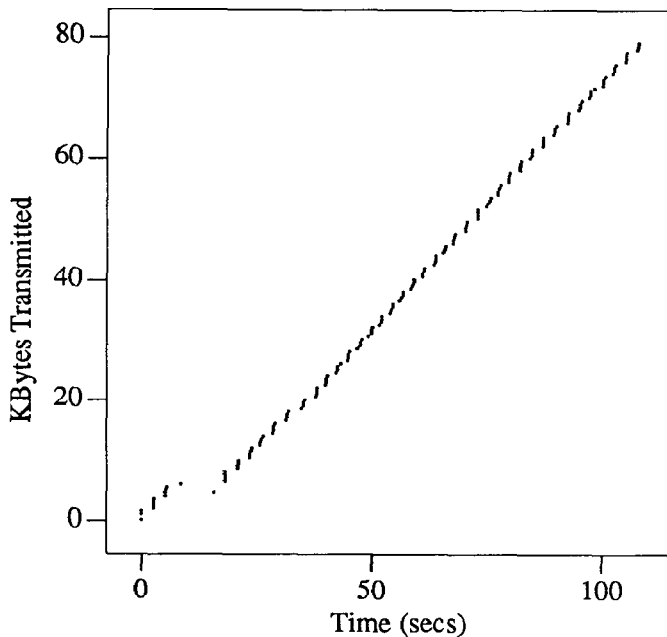


Figure 9: XTCP via SATNET-ARPANET path

Multiple connections - This experiment involved two TCP connections from a Sun on UCL LAN to a Sun on NTA-RE LAN. The TCP was XTCP with an MSS of 420 octets, and an advertised receive window of 8 Kbytes. Figure 10 shows how the congestion window algorithm shares bandwidth fairly among TCP connections.

7.5 Analysis of Results

A good transport protocol implementation in a connectionless Internet must adapt to loss, delay, load variations, and congestion.

Some initial TCP implementations had fixed timeouts and simplistic adaptive windows. Usually these were intended for a low delay, low error rate, uncongested LAN. However, these implementations were ill-suited to the SATNET environment. Current measurements over SATNET, either by itself or further on into ARPANET, show that the combination of changes made to the 4.3BSD TCP have vastly improved TCP performance, from a previous throughput of 3-4 Kbits/sec, to greater than 12 Kbits/sec. The most significant improvement is the reduction of the number of unnecessary retransmissions, which has effectively dropped to zero.

Recently, it has been observed that the characteristics of SATNET in terms of delay-bandwidth product match those of the

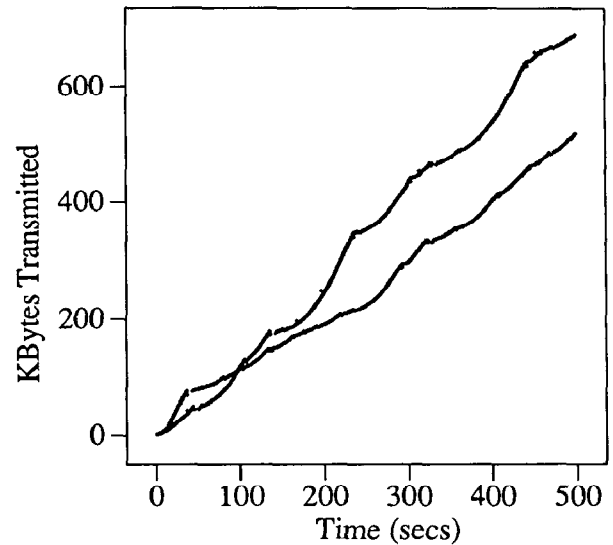


Figure 10: 2 XTCP Connections via SATNET

high speed experimental networks that are emerging as the new technology. Although SATNET is not identical to these networks, it should be an excellent starting place for investigating protocol behavior over such paths, since the actual data rates are slow enough that the quantity of data that must be collected to characterize such behavior is much lower than that required for a high speed, lower delay network. It is also easier to timestamp events accurately without the requirement for special clock hardware.

Future work might include the addition of a possible selective acknowledgement/retransmission option to TCP [1] to deal with SATNET's random loss characteristics. The TCP implementation that uses "slow start" and "congestion avoidance" waits for one RTT after loss before retransmitting, and one RTT further for a possible acknowledgement before opening the whole window or starting the slow start. Selective acknowledgements could be used to decouple detection of random loss from detection and avoidance of congestion so that packet loss due to errors did not result in unnecessary throttling of throughput.

8 Conclusions

8.1 Applicability of this work to the Internet

It is hoped that the experience gained by the SATNET Measurement Taskforce will apply not just to SATNET but to Internet design and analysis more generally. The tools and approach should be effective for analysis in other settings. Our results should provide clues to the behavior of protocols in other environments.

In particular, SATNET proved to be an ideal testbed for examining the performance of end-to-end protocols such as TCP/IP. In analyzing performance over SATNET, experiments were done that included many components of the Internet (Internet gateways, ARPANET, and local area networks at SATNET sites). The tools and experimental approach worked well in these tests and should generalize to the analysis of other Internet components and topologies.

One specific area that seems relevant to ongoing work in the Internet is the problem of finding the right window size for a 128 Kbits/sec network with a high round-trip delay (1 second). The SATNET case should be relevant to finding the right window for a higher bandwidth network with a lower round-trip delay. For example, the SATNET window could be as large as 128 Kbits/sec x 1 second. This is comparable to a high bandwidth network

with 100 Mbits/sec x .001 seconds. A more concrete example is the new NSFNET which will use 1.3 Mbit/sec T1 links. For a transcontinental round-trip time of about 50-100ms, the delay-bandwidth product is 65-130 Kbits, similar to SATNET's. Since this round-trip time is fixed, higher speed networks will also have this problem. It is important to note that in order to reliably handle the volume of data needed to fill the SATNET pipe, the SIMPs would need enough buffering for a data quantity 2-3 times the delay-bandwidth product.

Another relevant area is SATNET's load sharing algorithm. SATNET's ability to share bandwidth differently between different sites, both by the way in which a given channel is allocated and by the way in which traffic is allocated between the two channels, is similar to network architectures currently under consideration in which a packet switch must dynamically allocate traffic among multiple links to a given destination.

In general, protocol implementations originally tuned for low-loss, low-delay networks have not performed as well when used across a high bandwidth, high-delay, high-delay-variance network. As the Internet grows to include networks of even greater disparity and capability, the importance of implementations exhibiting stability and effective use of the internetwork resources increases.

8.2 Directions for Future Work

There are several directions in which this form of distributed test and measurement (T&M) could proceed.

The first direction is focused on specific issues of packet networks in the satellite communications environment. The suite of testing tools now developed could be used for additional study of the coupling between TCP/IP performance, traffic loading, link error and bandwidth, and SATNET link and management protocols.

Of particular interest would be determination of the optimal sets of SATNET protocols under varying traffic loadings, link error rates and use of one to many broadcast channels in the same network. Also interesting would be the conditions in which Wideband Network PODA [8] is more effective than SATNET PODA. A study could use link error rates varying across several orders of magnitude. This would be more like the harsher conditions in military satellite communications than the current benign environment of civil PTTs.

A second direction is to use and extend the T&M tools to study protocol performance issues in the broader Internet. These studies could be protocol specific, such as better tuning of individual TCP implementations, or could aim at better understanding a generic issue, e.g., IP fragmentation and reassembly schemes, or could straddle protocol layers and reveal their interaction, e.g., Telnet/FTP performance consequent upon that of TCP/IP.

Finally, these T&M tools, if more broadly implemented, could help establish performance standards for the Internet and aid the evaluation of new protocol developments and networking technologies. These tools should be seen as essential for the management of the operational Internet, and should be included in the protocol and management products offered by the vendor community.

8.3 Summary

In order to successfully measure SATNET performance, the taskforce:

- tested the components (protocol layer and part of the network path) separately
- defined the independent and dependent variables and identified the key parameters of the protocols involved
- built up a suite of tools - IP and TCP message generators, echoers, monitoring software

- followed an agreed-upon experimental procedure to ensure comparability and reproducibility of tests.

The IP-level service provided by SATNET had the following characteristics. Average round-trip delay was about 2.1 seconds with approximately 1% of the packets experiencing much higher delay. Maximum throughput was about 36 Kbits/sec for large packets and 17.2 Kbits/sec for small packets. Packet loss ranged from .1% to 1%.

The TCP-level service over this IP service showed that:

- The experimental 4.3 BSD TCP provided better throughput (12 Kbits/sec) than the old TCP (3-4 Kbits/sec). This was the result of changes to MSS selection, window sizing, RTT estimation, and retransmission strategy.
- The experimental 4.3 BSD TCP reduced the number of unnecessary retransmissions almost to 0.

SATNET proved to be an excellent test environment for examining end-to-end performance issues. In addition to providing built-in measurement tools, SATNET had characteristics that stressed end-to-end protocols - relatively high packet loss and high delay with high variability in round-trip delay.

ACKNOWLEDGEMENTS

Many people have made significant contributions to the efforts of the SATNET Measurement Taskforce and to this document. We would like to take this opportunity to acknowledge the work done by other members (past and present) of the taskforce - Peter Lloyd (University College London), Carl Rokitanski (The German Aerospace Research Establishment) and Claudio Topolic (Bolt Beranek and Newman), who did much of the early work on defining the test plans and on TCP/IP characterization of the network; and Håvard Hegna and Dan Grimstad, who developed the NTA-RE/NCC IP measurement tool and did many of the recent tests that helped define IP performance.

In addition, we would like to thank Ed Cain (Defense Communications Engineering Center) and Luciano Lenzini and Blasco Bonito (Italian National Research Council) for providing resources and support at their sites.

We would also like to thank Van Jacobson (Lawrence Berkeley Laboratory) for his insightful comments and help in examining our results. Our work has benefited greatly from his ideas and TCP/IP experimentation.

Lastly, we would like to acknowledge the technical and editorial assistance we've received from Winston Edmond (Bolt Beranek and Newman) and Charles Lynn (Bolt Beranek and Newman). Both have contributed their expertise to the measurements work as well as this document.

References

- [1] Braden R. and Jacobson V., "TCP Extensions for Long-Delay Paths", Draft RFC in preparation, Network Information Center, SRI International, Menlo Park, CA, 1988.
- [2] Chu W.W., Gerla M., Naylor W.E., Treadwell S., Mills D., Spilling P., Lundh Y. and Aagesen F.A., "Experimental Results on the Packet Satellite Network", *Proceedings of NTC*, Washington, Nov 1979.
- [3] Crowcroft, J., draft report of the SATNET Measurement Taskforce on TCP/IP performance over the SATNET, University College London, London, 1988
- [4] Davies, Brian H., and Bates, Andrew S., "Internetworking in the Military Environment", *Conference Proceedings of INFOCOM*, Las Vegas March 30-April 1, 1982, pp 19-29.
- [5] Easterbrook, Steve, "A Toolkit for Monitoring TCP", Department of Computer Science, University College London, Gower Street, London, June 1987.

- [6] Edmond W., Blumenthal S., Echenique A., Storch S., Calderwood T., and Rees T., "The Butterfly Satellite IMP for the Wideband Packet Satellite Network", *Proceedings of ACM SIGCOMM '86*, August 5-7, 1986.
- [7] Edge S.W., "An Adaptive Timeout Algorithm for Retransmission Across a Packet Switching Network", *Proc. of SIGCOMM '83*, ACM 1983.
- [8] Falk G., Groff S., Milliken W., Nodine M., Blumenthal S., and Edmond W., "A Multiprocessor Channel Scheduler for the Wideband Packet Satellite Network", Conference Record, IEEE International Conference on Communications, pp 9-16, Boston, MA, June 1983.
- [9] Gerla M. and Naylor W., "Final Technical Report of the Packet Satellite Program", Computer Science Department, UCLA, Los Angeles, Nov 1978.
- [10] Hegna, H. Grimstad, T., Norwegian Computing Center and Spilling P., Grimstad D., Ryen J.; Norwegian Telecommunications Administration Research Establishment, "SATNET Measurements, Summer and Fall 1987", Norwegian Computing Center, Jan 1988.
- [11] Jacobs I., *et al.*, "CPODA-A demand assignment protocol for SATNET", *Proc. Data Commun. Symp. ACM, IEEE*, Sept. 1977, pp. 2-5-2-9.
- [12] Jacobs I.M., Binder R. and Hoversten E.V., "General Purpose Packet Satellite Network", *Proc. IEEE*, vol 66, pp. 1448 - 1467, Nov. 1978
- [13] An experimental version of TCP, based on UNIX 4.3 and modified for the adaptive retransmission strategy and "slow-start", was obtained from Van Jacobson.
- [14] Jain R., "A Timeout-based Congestion Control Scheme for Window Flow-controlled Networks", *IEEE J. Selected Areas in Comms.*, Vol SAC-4, No. 7, Oct 1986.
- [15] Leddy, J., draft report of the SATNET Measurement Taskforce on the final IP characterization of the SATNET, Bolt Beranek and Newman, Cambridge, Mass. 1988
- [16] Leiner B., Cole R., Postel J., Mills D., "The DARPA Internet Protocol Suite", *IEEE Communications Magazine*, Vol. 23, No 3, March 1985
- [17] Linkabit Corp., "Final Report for the Atlantic Packet Satellite Program", Tech. Rep., Linkabit Corp., Sept. 1980.
- [18] Mills D., "Internet Delay Experiments", RFC 889, Network Information Center, SRI International, Menlo Park, CA, 1983.
- [19] Nagel, "Congestion Control in IP/TCP", RFC 896, Network Information Center, SRI International, Menlo Park, CA, 1984.
- [20] Seo, K., draft report of the SATNET Measurement Taskforce on the methodology and tools that were used, Bolt Beranek and Newman, Cambridge, Mass. 1988
- [21] "Transmission Control Protocol, DARPA Internet Program Protocol Specification", RFC 793, Network Information Center, SRI International, Menlo Park, CA, September 1981.