

\*\*\*\*\*

ATM Forum Document Number: ATM\_Forum/97-0616

\*\*\*\*\*

Title: UBR Buffer Requirements for TCP/IP over Satellite Networks

\*\*\*\*\*

Abstract: In this contribution, we present simulation results to assess buffer requirements for TCP/IP over satellite UBR networks. We perform experiments with both LEO and GEO satellite delays, for various buffer sizes and number of sources. We conclude that with sufficiently large buffers (0.5RTT or more) the performance of TCP-SACK over UBR with per-VC accounting is scalable with respect to the number of sources.

\*\*\*\*\*

Source:

Rohit Goyal, Raj Jain, Sonia Fahmy, Bobby Vandalore, Shiv Kalyanaraman  
Department of CIS, The Ohio State University (and NASA)  
395 Dreesse lab, 2015 Neil Ave,  
Columbus, OH 43210-1277  
Phone: 614-292-3989, Fax: 614-292-2911, Email: {goyal,jain}@cis.ohio-state.edu

Sastri Kota  
Lockheed Martin Telecommunications/Astrolink Broadband Network Lab  
1272 Borregas Avenue, Samsung Electronics Co. Ltd.  
Bldg B/551 O/GB - 70 Samsung Telecom America, Inc.  
Sunnyvale, CA 94089 1130 E Arapaho, Richardson, TX 75081  
Email: sastri.kota@lmco.com email: psamudra@telecom.sna.samsung.com

\*\*\*\*\*

Date: July 1997, Montreal

\*\*\*\*\*

Distribution: ATM Forum Technical Working Group Members (AF-TM)

\*\*\*\*\*

Notice: This contribution has been prepared to assist the ATM Forum. It is offered to the Forum as a basis for discussion and is not a binding proposal on the part of any of the contributing organizations. The statements are subject to change in form and content after further study. Specifically, the contributors reserve the right to add to, amend or modify the statements contained herein.

\*\*\*\*\*

## 1 Introduction

Satellite communication systems play an important role in the integration of networks of various types and services. Satellite systems will be used for a wide range of applications and will play an important role in the future of the Global Information Infrastructure. The main advantages of satellite systems are the long range broadcasting ability, support of mobile systems, and potentially high available bandwidth. However, satellite systems have several inherent constraints. The resources of the satellite communication network, especially the satellite and the earth station have a high cost and must be used efficiently. *A crucial issue is that of the high end-to-end propagation delay of satellite connections.*

The ATM-UBR service category is relatively cheap to implement in switch hardware. As a result, switches can multiplex thousands of transport connections that use the UBR service for non-real time applications. On board satellite switches and switches at the earth stations fall into this category and are expected to multiplex a large number of transport connections over UBR virtual circuits.

Apart from interoperability issues, several performance issues need to be addressed before a transport layer protocol like TCP can satisfactorily work over UBR. Moreover, with an acknowledgment and timeout based congestion control mechanism (like TCP's), the performance is inherently related to the delay-bandwidth product of the connection. As a result, the congestion control issues for high bandwidth satellite networks can be somewhat different from those of LAN and WAN networks. The performance optimization problem can be analyzed from two perspectives – network policies and end system policies. The network can implement a variety of mechanisms to optimize resource utilization, fairness and higher layer throughput. For UBR, these include enhancements like intelligent drop policies to improve utilization, some minimal per-VC accounting [1, 2] to improve fairness, and even minimum throughput guarantees to the higher layers.

At the end system, the transport layer can implement various congestion avoidance and control policies to improve its performance and to protect against congestion collapse. Several transport layer congestion control mechanisms have been proposed and implemented. The mechanisms implemented in TCP are slow start and congestion avoidance [5], fast retransmit and recovery [7], and selective acknowledgments [8]. Several others like forward acknowledgments [9] and negative acknowledgments [4] have been proposed as enhancements to timeout based schemes.

Studies have shown that small switch buffer sizes result in very low TCP throughput over UBR [2]. It is also clear, that the buffer requirements increase with increasing delay-bandwidth product of the connections (provided the TCP window can fill up the pipe). However, the studies have not quantitatively analyzed the effect of buffer sizes on performance. *As a result, it is not clear how the increase in buffers affects throughput, and what buffer sizes provide the best cost-performance benefits for TCP/IP over UBR.*

In this contribution, we present our simulation results to assess the buffer requirements for various delay-bandwidth products for TCP/IP over UBR.

## 2 Previous Work: TCP performance over UBR

In our previous work, we have studied TCP performance over the ATM-UBR service for LAN, WAN and satellite networks. In our studies, we have used an N-source symmetrical TCP configuration with unidirectional TCP sources. The performance of TCP over UBR is measured by the efficiency and fairness which are defined as follows:

$$\text{Efficiency} = (\text{Sum of TCP throughputs}) / (\text{Maximum possible TCP throughput})$$

The TCP throughputs are measured at the destination TCP layers. Throughput is defined as the total number of bytes delivered to the destination application, divided by the total simulation time. The results are reported in Mbps.

The maximum possible TCP throughput is the throughput attainable by the TCP layer running over UBR on a 155.52 Mbps link. For 9180 bytes of data (TCP maximum segment size), the ATM layer receives 9180 bytes of data + 20 bytes of TCP header + 20 bytes of IP header + 8 bytes of LLC header + 8 bytes of AAL5 trailer. These are padded to produce 193 ATM cells. Thus, each TCP segment results in 10229 bytes at the ATM Layer. From this, the maximum possible throughput =  $9180/10229 = 89.7\%$  = 135 Mbps approximately on a 155.52 Mbps link (149.7 Mbps after SONET overhead).

$$\text{Fairness Index} = (\sum x_i)^2 / (n \times \sum x_i^2)$$

Where  $x_i$  = throughput of the  $i$ th TCP source, and  $n$  is the number of TCP sources. The fairness index metric applies well to our N-source symmetrical configuration.

In most cases, the performance of TCP over UBR has been poor. A summary of our previous results is presented below [2, 3]:

- TCP achieves maximum possible throughput when no segments are lost. To achieve zero loss for TCP over UBR, switches need buffers equal to the sum of the receiver windows of all the TCP connections.
- With limited buffer sizes, TCP performs poorly over vanilla UBR switches. TCP throughput is low, and there is unfairness among the connections. The coarse granularity TCP timer is an important reason for low TCP throughput.
- Efficiency typically increases with increasing buffer size.
- Fast retransmit and recovery improve performance for LAN configurations, but degrade performance in long latency configurations.
- SACK TCP improves performance especially for large latency networks.
- Early Packet Discard improves efficiency but not fairness.
- Per-VC buffer management improves both efficiency and fairness.

### 3 Buffer Requirements Study

In this contribution we present results of TCP throughput over satellite UBR for various delays, buffer sizes and number of sources.

1. **Latency.** Our primary aim is to study the performance of large latency connections. The typical latency from earth station to earth station for a single LEO (700 km altitude, 60 degree elevation angle) hop is about 5 ms [10]. The latencies for multiple LEO hops can easily be up to 50 ms from earth station to earth station. GEO latencies are typically 275 ms from earth station to earth station. We study these three latencies (5 ms, 50 ms, and 275 ms) with various number of sources and buffer sizes.
2. **Number of sources.** To ensure that the recommendations are scalable and general with respect to the number of connections, we will use configurations with 5, 15 and 50 TCP connections on a single bottleneck link. For single hop LEO configurations, we use 15, 50 and 100 sources.
3. **Buffer size.** This is the most important parameter of this study. The set of values chosen are  $2^{-k} \times RTT$ ,  $k = -1.6, 2, 1, 0.5, 0.25, 0.125, 0.0625, 0.031, 0.016$  multiples of the round trip delay-bandwidth product of the TCP connections.) We plot the buffer size against the achieved TCP throughput for different delay-bandwidth products and number of sources. The asymptotic nature of this graph provides information about the optimal buffer size for the best cost-performance ratio.
4. **Switch drop policy.** We use a per-VC buffer allocation policy called selective drop (see [2]) to fairly allocate switch buffers to the competing connections.
5. **End system policies.** We use an enhanced version of TCP called SACK TCP for this study. SACK TCP improves performance by using selective acknowledgements for retransmission. Further details about our SACK TCP implementation can be found in [3].

### 4 Simulation Setup

Figure 1 shows the basic network configuration that was simulated. In the figure, the switches represent the earth stations that connect to the satellite constellation. The entire satellite network is assumed to be a 155 Mbps ATM link without any on board processing or queuing. All processing and queuing are performed at the earth stations.

- All simulations use the N source configuration shown in Figure 1. All sources are identical and infinite TCP sources. The TCP layer always sends a segment as long as it is permitted by the TCP window. Moreover, traffic is unidirectional so that only the sources send data. The destinations only send ACKs. The delayed acknowledgement timer is deactivated, and the receiver sends an ACK as soon as it receives a segment. As discussed before, SACK TCP is used in our simulations.

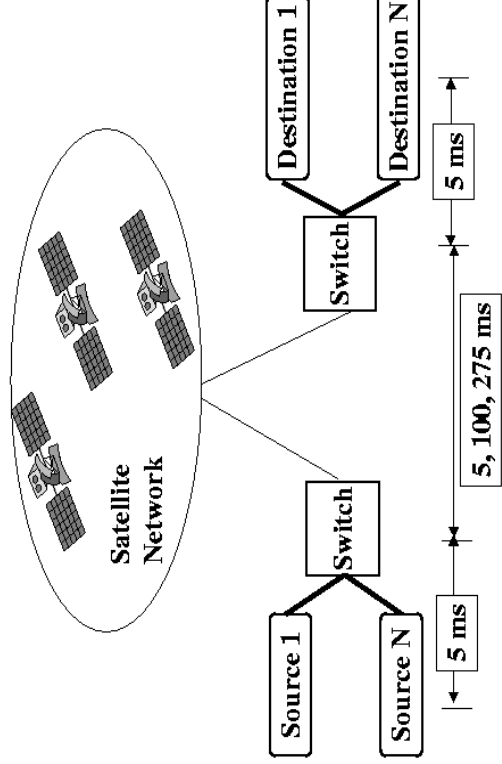


Figure 1: The N source TCP configuration

- Three different configurations are simulated that represent a single LEO hop, multiple LEO hops and a single GEO hop. The link delays between the switches and the end systems are 5 ms in all configurations. The inter-switch (earth station to earth station) propagation delays are 5 ms, 100 ms, and 275 ms for single hop LEO, multiple hop LEO and GEO configurations respectively. This results in a round trip propagation delays of 30 ms, 120 ms and 570 ms respectively.
- The number of sources (N) was 15, 50, and 100 for single hop LEO, and 5, 15 and 50 for GEO and multiple hop LEO configurations.
- The maximum value of the TCP receiver window is 600000 bytes, 2500000 bytes and 8704000 bytes for single hop LEO, multiple hop LEO and GEO respectively. These window sizes are sufficient to fill the 155.52 Mbps links.
- The TCP maximum segment size is 9180 bytes. A larger value is used because most TCP connections over ATM with satellite delays are expected to use larger segment sizes.
- The buffer sizes (in cells) used in the switch are the following:
  - Single LEO: 375, 750, 1500, 3000, 6000, 12000 (=1 RTT), 24000 and 36000.
  - Multiple LEO: 780, 1560, 3125, 6250, 12500, 50000 (=1 RTT), and 100000.
  - GEO: 3375, 6750, 12500, 25000, 50000, 100000, 200000 (=1 RTT), and 400000.
- The duration of simulation is 100 seconds for multiple hop LEO and GEO and 20 secs for single hop LEO.
- All link bandwidths are 155.52 Mbps, and peak cell rate at the ATM layer is 14957 Mbps after the SONET overhead.

## 5 Simulation Results

Figures 2, 3, and 4 show the resulting TCP efficiencies for the 3 different latencies. Each point in the figure shows the efficiency (total achieved TCP throughput divided by maximum possible throughput) against the buffer size used. Each figure plots a different latency, and each set of points (connected by a line) in a figure represents a particular value of  $N$  (the number of sources). The following conclusions can be drawn from the figures:

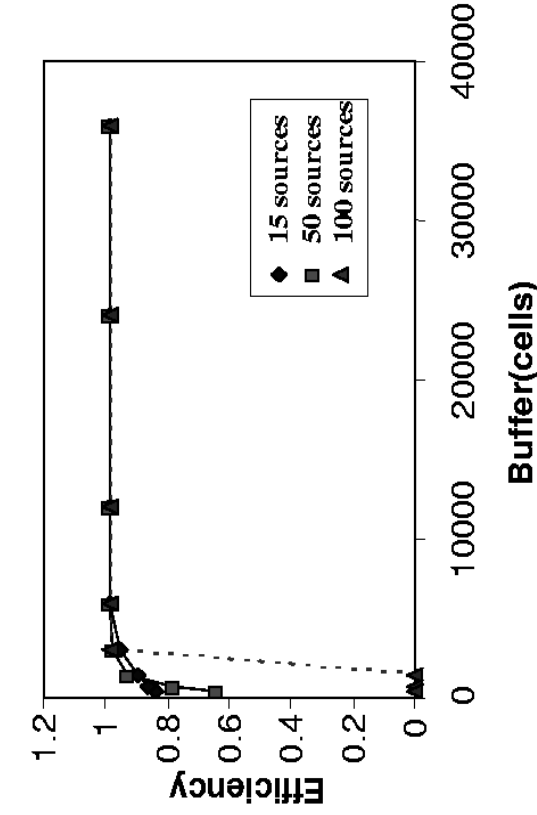


Figure 2: Buffer requirements for single hop LEO

1. For very small buffer sizes, ( $0.016 \times \text{RTT}$ ,  $0.031 \times \text{RTT}$ ,  $0.0625 \times \text{RTT}$ ), the resulting TCP throughput is very low. In fact, for a large number of sources ( $N=50$ ), the throughput is sometimes close to zero.
2. For moderate buffer sizes (less than 1 round trip delay-bandwidth), TCP throughput increases with increasing buffer sizes.
3. TCP throughput asymptotically approaches the maximal value with further increase in buffer sizes.
4. **TCP performance over UBR for sufficiently large buffer sizes is scalable with respect to the number of TCP sources.** The throughput is never 100%, but for buffers greater than  $0.5 \times \text{RTT}$ , the average TCP throughput is over 98% irrespective of the number of sources.
5. The knee of the buffer versus throughput graph is more pronounced for larger number of sources. For a large number of sources, TCP performance is very poor for small buffers, but jumps dramatically with sufficient buffering and then stays about the same. For smaller number of sources, the increase in throughput with increasing buffers is more gradual.

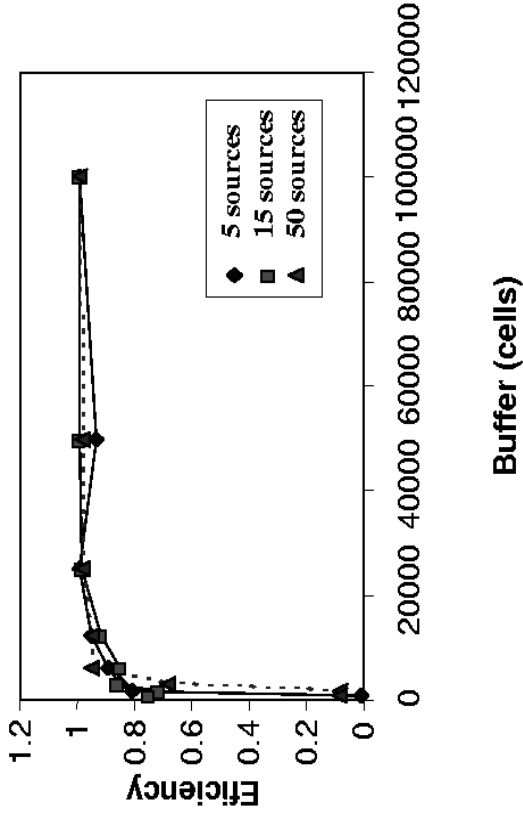


Figure 3: Buffer requirements for multiple hop LEO

6. **For large round trip delays, and a small number of sources, a buffer of 1 RTT or more can result in a slightly reduced throughput.** This is because of the variability in the TCP retransmission timer value. When the round trip is of the order of the TCP timer granularity (100 ms in this experiment), and the queuing delay is also of the order of the round trip time, the retransmission timeout values become very variable. During the initial phase (startup exponential increase), when the queuing delays are small, the timeout value corresponds to the propagation delay. When the windows increase to fill the switch buffer, the queuing delay increases to about 1 RTT and packets at the tail of the queue get dropped. Retransmitted packets are sent out after 3 duplicate ACKS are received. However, these retransmitted packets are queued behind a whole RTT worth of queues at the bottleneck switch. As a result, before the sender gets an ACK for retransmitted packets, a timeout occurs, and slow start is incurred. At this point, the sender starts to retransmit from the last unacked segment, but soon receives an ACK for that segment (because the segment was not really lost, but the delay was incorrectly estimated). The loss in throughput is due to the time lost in waiting for the retransmission timeout.
7. Fairness is high for a large number of sources. This shows that TCP sources with a good per-VC buffer allocation policy like selective drop, can effectively share the link bandwidth.

## 6 Summary

A buffer size of about  $0.5 \times \text{RTT}$  to  $1 \times \text{RTT}$  is sufficient to provide over 98% throughput to infinite TCP traffic for long latency networks and a large number of sources. This buffer requirement is independent of the number of sources. The fairness is high for a large number of sources because of the nature of TCP traffic and the per-VC buffer management performed at the switches.

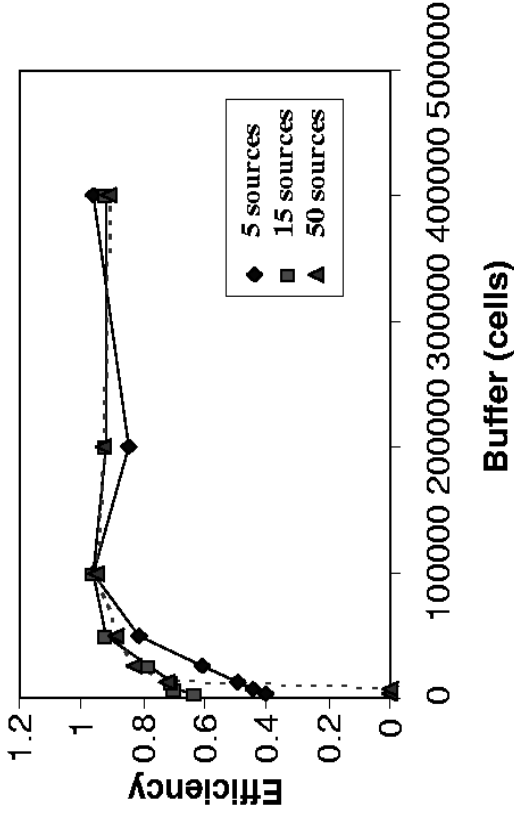


Figure 4: Buffer requirements for GEO

Throughput may slightly decrease for buffers larger than 1RTT because of variability in the RTT estimate approaches the timer granularity.

## References

- [1] Juha Heinanen, and Kalevi Kilkki, "A fair buffer allocation scheme," Unpublished Manuscript.
- [2] R. Goyal, R. Jain, S. Kalyanaraman, S. Fahmy and Seong-Cheol Kim, "UBR+: Improving Performance of TCP over ATM-UBR Service," Proc. ICC'97, June 1997.
- [3] R. Goyal, R. Jain et.al., "Selective Acknowledgements and UBR+ Drop Policies to Improve TCP/UBR Performance over Terrestrial and Satellite Networks," To appear: Proceedings of IC3N'97, September 1997.<sup>1</sup>
- [4] Grace Yee, Sastri Kota, Gary Ogasawara, "TCP Performance over a Satellite ATM Network," Submitted to Globecom'97.
- [5] V. Jacobson, "Congestion Avoidance and Control," Proceedings of the SIGCOMM'88 Symposium, pp. 314-32, August 1988.
- [6] V. Jacobson, R. Braden, "TCP Extensions for Long-Delay Paths," Internet RFC #072, October 1988.
- [7] V. Jacobson, R. Braden, D. Borman, "TCP Extensions for High Performance," Internet RFC 1323, May 1992.

<sup>1</sup> All our papers and ATM Forum contributions are available from <http://www.cis.ohio-state.edu/~jain>

- [8] M. Mathis, J. Madhavi, S. Floyd, A. Romanow, “TCP Selective Acknowledgement Options,” Internet RFC 2018, October 1996.
- [9] M. Mathis, J. Madhavi, “Forward Acknowledgement: Refining TCP Congestion Control,” Proc. SIGCOMM’96, August 1996.
- [10] S $\Sigma$ tellite altitudes taken from “Lloyd’s satellite constellation,” <http://www.ee.surrey.ac.uk/Personal/L.Wood/constellations/overview.html>