Performance limitations due to TCP Burstiness in GEO satellite networks with limited buffering.

Mahesh Sooriyabandara[§], Godred Fairhurst[§]

[§] Electronics Research Group, University of Aberdeen. UK.

{mahesh,gorry}@erg.abdn.ac.uk

Abstract: *Next generation GEO satellites will use an on board switch to facilitate efficient telecommunication networks. They will provide Internet service provision to particular groups of users. Therefore it is reasonable to believe that, major portion of data traffic to be based on TCP protocol. TCP is bursty in nature. Burstiness of TCP can result poor performance due to the limited buffering available in the on board switch. Several researchers have identified the various problems associated with TCP bursts and have proposed solutions. In this paper we try to illustrate the problem of TCP burstiness in a GEO satellite network using experimental results. Currently we are analyzing the scenario in a simulator aiming to design a transparent support mechanism to alleviate the problem.*

1. Introduction.

TCP [1] is the most commonly used transport protocol in the Internet. It was initially designed to suit different kinds of network conditions. However, modern networking and access technologies in satellite and high-speed communications have created new challenges for TCP. Next generation GEO satellites will be much more advance and complex than the legacy bent pipe networks. They will carry an on board switch to form a mesh type network. These networks will have a high capacity. Apart from the well-known problems due to high delay and high capacity of satellite paths, the limited buffer capacity of on board switch will impose another challenge for TCP performance.

In a high delay bandwidth network a TCP session needs to have a large enough window [2] to fully utilize the link bandwidth. TCP maintains its send rate by dumping “cwnd”[1] worth of data during a RTT (round trip time)[1]. TCP implementations originally expected to follow a property known as “Self-Clocking”[3]. Self-Clocking means the temporal ACK spacing which reflects the bottleneck bandwidth in the path. However due to heterogeneity of modern networks most believes that this property is hardly preserved or in the worst case can not be seen. Therefore, the “cwnd” worth of packets are not evenly space through out the RTT period. Large bursts of data added to the network in short interval tend to create long queues in intermediate routers. In the worst case if window synchronization [4] occurs large bursts belong to different flows can overshoot the router buffers causing multiple drops. The bursty nature of TCP traffic has been identified as a common problem in different kinds of networks (e.g. bandwidth asymmetric networks[10]). In this paper we try to illustrate the effects of TCP bursts in a high bandwidth delay network with insufficient buffering.

2. TCP burstiness.

TCP traffic flows can show burstiness in two different ways [6]. First is a sort of micro burstiness. This is caused by the number of segments send in respond to each ACK received. The second could termed as a “macro burstiness”, which caused by ACKs arriving one after another causing a large amount of data transmission. This can be significant in high bandwidth delay paths. During slow start all ACKs arriving at the beginning of the RTT, causing a macro burst of data packets followed by a long idle period waiting for new ACKs. Apart from these TCP bursts can arise from some other causes like ACK congestion or due to ACK Filtering effects [10].

Problems created by TCP bursts

- **Increase buffer requirements. :** TCP optimizes its send rate by dumping increasingly large bursts, one burst per RTT until it reaches its maximum window size. To fill up a high capacity pipe TCP need to use a large maximum window. In most practical cases, the maximum size of window, which reflects the largest possible size of traffic burst, is very much higher than the queuing capacity of any intermediate router. Once TCP senders over load the router queues they will start to drop packets. TCP will see these packet drops due to queuing bottleneck as network congestion. This can result poor TCP performance like low throughput and unfair sharing. [5] describes the minimum buffer requirement for a TCP flow to fully utilize the link. When consider a TCP connection with out "ACK delay" [1], in ideal situation ACK rate represents the bottleneck rate. During “Slow Start” sender will send twice as much as the number it received during the RTT. Therefore ideally router buffer should at least be half the maximum window size ($W_{max}/2$). This is because during first half of the RTT, router receives as twice the rate as it sends and so it drains the queue during the second half of the queue. With same logic with ACK_delay minimum queue size should at least be one third of the maximum window size ($W_{max}/3$) [5].
- **Increased queuing delay:** Bursty nature of traffic leads to buffering of packets at the intermediate routers. However, buffering delays could also depend on the congestion level, queuing and scheduling policies. Larger queues at the routers may introduce additional delays to the TCP flows and increase their RTT. TCP applications like web browsing, file transferring etc. do not need stringent delay requirements. However increased round trip times can slowdown the TCP “cwnd” growth during slow start causing low throughput levels [7].
- **Jitter (variable delay):** In a GEO network delay variation mainly occurs due to queuing resulting from bursty traffic. TCP flows are robust in the presence of jitter but to identify the real effects further investigations are needed [7]. How ever the jitter and variation of jitter can be a crucial issue when other delay sensitive traffic share the network with bursty TCP traffic.
- **Making other traffic flows bursty:** The burstiness of one TCP flow can cause the other TCP flows to become bursty when they share a common queue at an intermediate router. The effect can be catastrophic if these bursts get synchronized. According to Global synchronization effect described in [4] packet losses due to buffer over flows can be synchronized causing all the TCP flows to back off simultaneously under utilizing the bandwidth.
- **Low throughput levels:** Burstiness of TCP would results packet drops arising from queue overflows or an increase RTT due to queuing delays. These will leads to low throughput levels of TCP flows. This along with other side effects like synchronized window evolution could waste the bandwidth.
- **Unfair sharing:** Busty nature of TCP may result unfairness among competing traffic because of the queuing bottleneck.

3. Modeling and analyzing.

A set of experiments was carried out in an emulated test bed to understand the effect of TCP burstiness with insufficient buffering. Test bed consists of two end hosts routed via a PC, which emulates the GEO satellite path. "dummynet" [8] was used to emulate the network. End system TCP implementations were TCP NewReno [9]. Further, TCP large window extensions [2] were enabled to achieve full bandwidth utilization (i.e. send and recv buffers were set to 150Kbytes). Network was configured with 2Mbps forward and return bandwidth and the total delay was set to 600ms. MSS [1] was 1460bytes and ACK delay was enabled [1]. Netperf [8] was used to generate the TCP traffic. Following graphs are presented to illustrate the bursty nature of TCP.

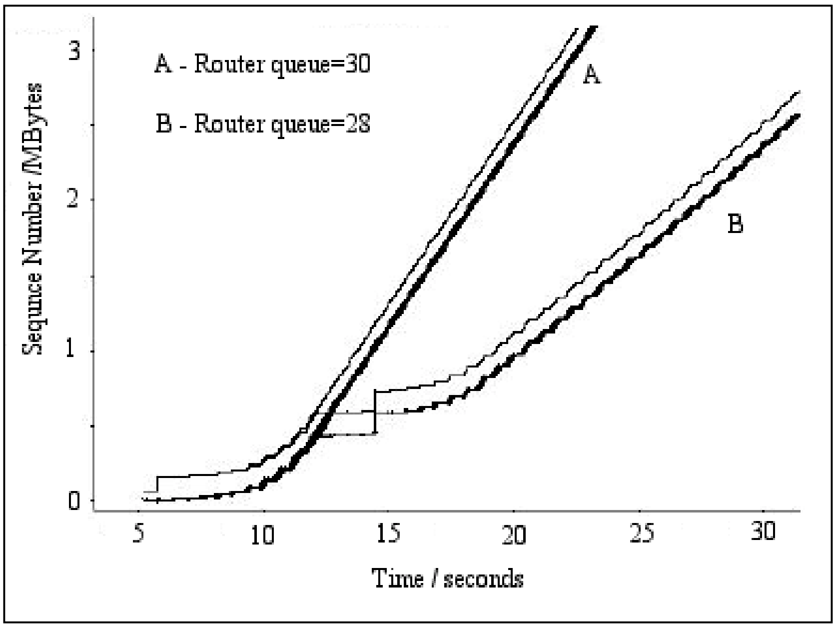

Figure 1 TCP seuqence plot.

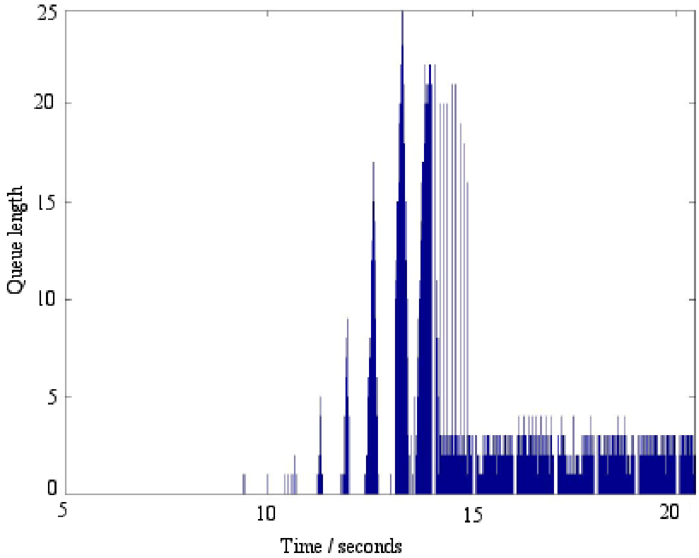

Figure 2 Variation of Router queue length Vs time.

The Figure 1 above shows a single TCP session. In case A router buffer was set to 30 packets. This value is nearly equal to the W_{mac} /3 which is the theoretical minimum router buffer requirement. It can be seen that, a TCP session is extremely burst during Slow Start period. As long as the buffering is sufficiently large it can maintain a higher throughput level. Curve B represents a case where insufficient buffering degrades TCP performance. In this case packet drops due to queuing bottleneck at the router drops the session back to Slow Start. A premature ending of slow start can be seen since buffering capacity is much lower than the minimum requirement.

Figure 2 shows the queuing of packet due to burstiness of TCP traffic. Exponential growth of the router queue during initial period represents the macro burstiness of TCP during Slow Start. Much lower level of queue occupancy during later part of the session's life occurs due to the micro burstiness of TCP.

4. Summary.

Burstiness is an inherited characteristic of TCP. The bursty nature of traffic can degrade the overall network performance in different forms. The effect can be critical in a network with insufficient buffering. Various researchers studying different network technologies have identified this problem. Some of the proposed solutions are namely TCP pacing [5], ACK spacing [11], various TCP rate control mechanisms and cascading TCP connections. Our aim is to investigate for feasible transparent mechanisms to improve TCP performance in a GEO satellite network with on board switching capabilities.

References.

[1] W.R.Stevens," TCP/IP illustrated- The protocols " Vol1 Addison Wesley, New York 1994.

[2] V. Jacobson " Large window extensions to TCP" RFC 1323, 1992, IETF

[3] V. Jacabson "Congestion Avoidance and Control " SIGCOMM '88 ACM USA 314-329 (1988).

[4] L. Zhang, S, Shenker and D.D. Clark " Observations on the dynamics of a Congestion control algorithm: The effects of two-way traffic". In proc. Of ACM SIGCOMM'91, pp. 133-147,September 1991.

[5] Joanna Kulik, Robert Coulter, Dennis Rockwell, and Craig Partridge, "A Simulation Study of Paced TCP," BBN Technical Memorandum No. 1218, August 12, 1999.

[6] M Allman E mail sent to the TCPSAT mailing list, "Burstiness with increased CWND" Feb 2000.

[7] Rohit Goyal, Sastri Kota, Raj Jain, Sonia Fahmy, Bobby Vandalore, Jerry Kallaus, "Analysis and Simulation of Delay and Buffer Requirements of Satellite-ATM Networks for TCP/IP Traffic," Submitted to IEEE Journal of Selected Areas in Communications, March 1998.

[8] S. Parker C. Cshmechel " Some tests tools for TCP implementers" RFC 2398 IETF August 1998.

[9] Floyd, S. and T. Henderson, "The New Reno Modification to TCP's Fast Recovery Algorithm", RFC 2582, April 1999.

[10] H. Balakirshnan, V.N Padmanabhan, R. H. Katz "The Effects of Asymmetry on TCP Performance" Proc. of 3rd ACM Conference on Mobile Computing and Networking.

[11] C. Partridge "ACK spacing for high delay-bandwidth paths with in-sufficient buffering." Draft-patridge-e2e-ackspacing-00.txt July 1997.