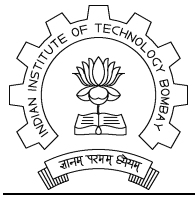

NS2 Simulation of TCP over satellite links

Sarvjeet Singh, 99005029

Problem Statement and Motivation

This project involves simulation of TCP over satellite links. This simulation will be done using NS2 package. TCP/IP is the protocol used as the transport mechanism for most file and data transfers in the internet. Understanding the perormance of TCP is important because it is the dominant protocol used in the Internet today. The characteristics of the underlying transport services have a direct effect on the performance of protocols at the higher layers, particularly if some of the transport links have higher bit error rates or higher latency than the protocol designer may have envisioned. Satellites have been used for data communications for many years and is a important medium for data transfer over the globe in present time. Satellite links are characterized by high propagation delay, high bandwidth and often, asymmetric bandwidth. If we want to use the existing TCP/IP protocol over satellite links then its desirable that we analyse its performance and note the effect of changing the parameters of TCP/IP protocol so as to discover the optimal values for these parameters. This kind of analysis can also help us to improve upon the current protocol by giving us insight into the bottlenecks of the protocol over satellite links. So we want to study effect of such kind of link on the performance of different types of applications running on TCP, namely, interactive (telnet etc.) and bulk transfer (FTP etc.) using the simulation package - NS2 simulator.

There are many mechanisms that may improvethe performance of TCP protocol over a given media. Some of the important ones are

- **Slow Start and Congestion Avoidance**: Slow start begins by sending one segment and waiting for an acknowledgement. For each acknowledgement the sender receives, it injects two segments into the network; leading to an exponential increase in the amount of data being sent. Slow start ends when the receiver's advertised window is reached or when the loss is detected. Because the amount of time required for slow start to achieve full bandwidth is a function of round trip timem, satellite links will be particularly sensitive to the limited throughput available during slow start.

  Congestion avoidance is used to prove the netword for available bandwidth by sending one additional segment for each round trip time. When the TCP detects segment loss, which indicates congestion, it drops back into slow start until the packet sending rate is half the rate at which the loss was detected and then begins the congestion avoidance phase.

- **Fast Retransmit and Fast Recovery**: In fast retransmit, rather than waiting for retransmit timeout (RTO), the TCP sender can retransmit a segment if it receives three duplicate ACKs for the segment sent immediately before the lost segment. This reduces

the time it takes a TCP sender to detect a single dropped segment. Fast recovery works hand in hand with fast retransmit. If sending TCP detects a segment loss using fast retransmit, fast recovery is used instead. Fast receovery halves the segment sending rate and begins cngestion avoidance immediately, without falling back to slow start.

- **Large Windows**: The orginal TCP standard limits the TCP receive window to 65535 bytes. TCP's receiving window size is particularly important in a satellite environment becuase the maximum throughput of a TCP connection is bounded by the round trip time. Large windows can allow TCP to fully utilize higher bandwidth links over long-delay channels suchas those found in satellite links.

- **Selective Acknowledgements** The cumulative positive acknowledgements employed by TCP are not particularly well suited to the long-delay satellite environment due to the time it takes to obtain information about segment loss. A selective acknowledgement (SACK) mechanism defined in RFC 2018. SACK's generated at the receiver explicity inform the sender about which segments have arrived and which may have been lost, giving the sender more information about which segments might need to be retransmitted.

Model for Analysis and Implementation

I plan to use the satellite extension in NS2 to simulate this project. I will have two nodes with a satellite link between them. I will then simulate and find out the performance of the network by changing different parameters of the TCP protocol and by introducing the schemes mentioned above (and it time permits other methods like delayed ACKs and Cumulative ACKs) and find out how far they help improve the performance over satellite link for interactive transfer and bulk transfer. The behavior of links can be observed by constructing a sequence number Vs Time plot and measuring the throughput.