ENSC 835 project (2002) TCP performance over satellite links

Kenny, Qing Shao
Qshao@cs.sfu.ca
Grace, Hui Zhang
Hzhange@cs.sfu.ca

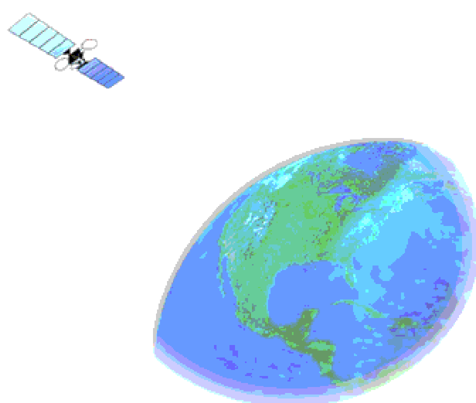
communication networks laboratory

Road map

- Introduction to satellite communications
- Simulation implementation
  - Window size
  - Maximum segment size
  - Initial windows size
  - Selective acknowledgement algorithm
  - TCP burst
- Conclusion
- References

Introduction to satellite communications

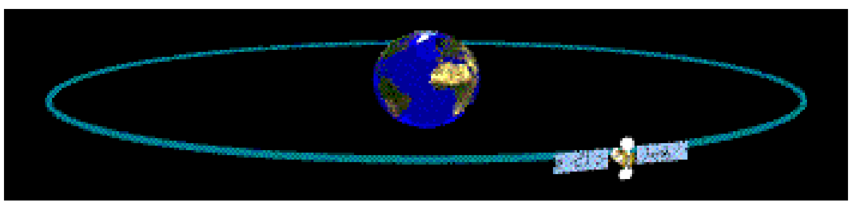
- Satellite communication is a type of the wireless communications technologies. It utilizes satellites to retransmit the wireless signal, and to connect with the multiple earth station.

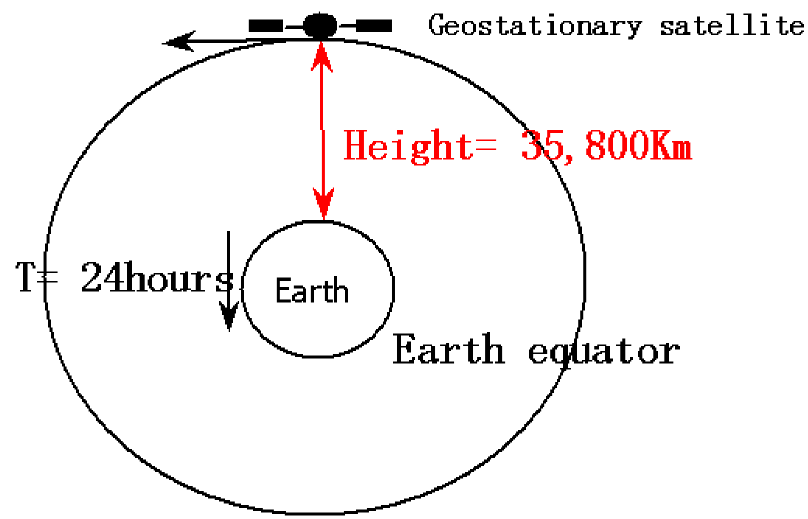


Introduction to satellite communications

Geosynchronous Equatorial Orbit

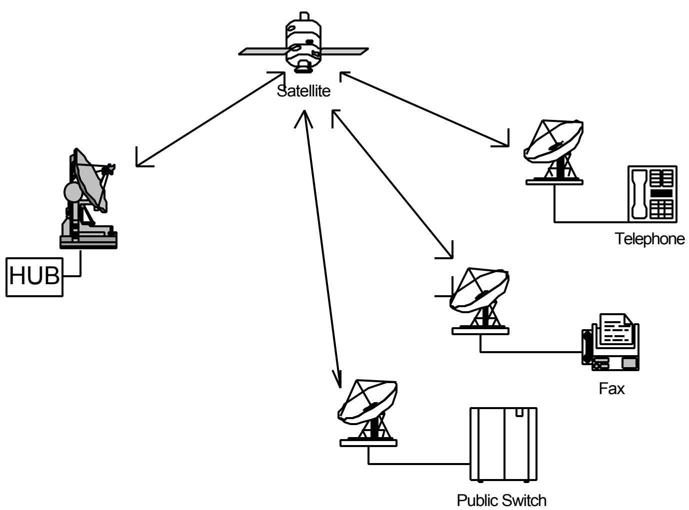
(From ***geo*** = earth + ***synchronous*** = moving at the same rate.)

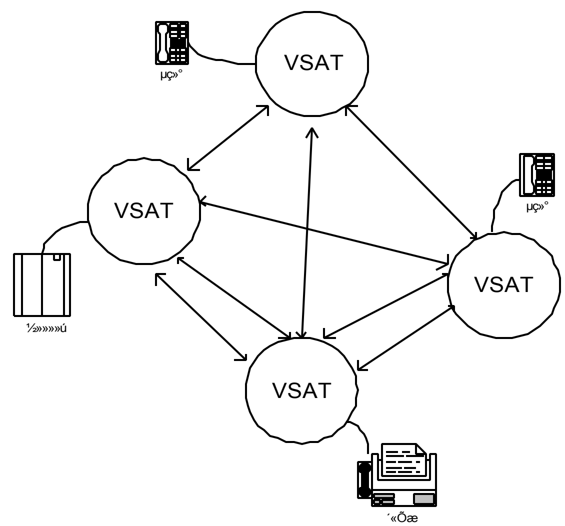




On board GEO satellite

Bent pipe GEO satellite | On board switch GEO satellite

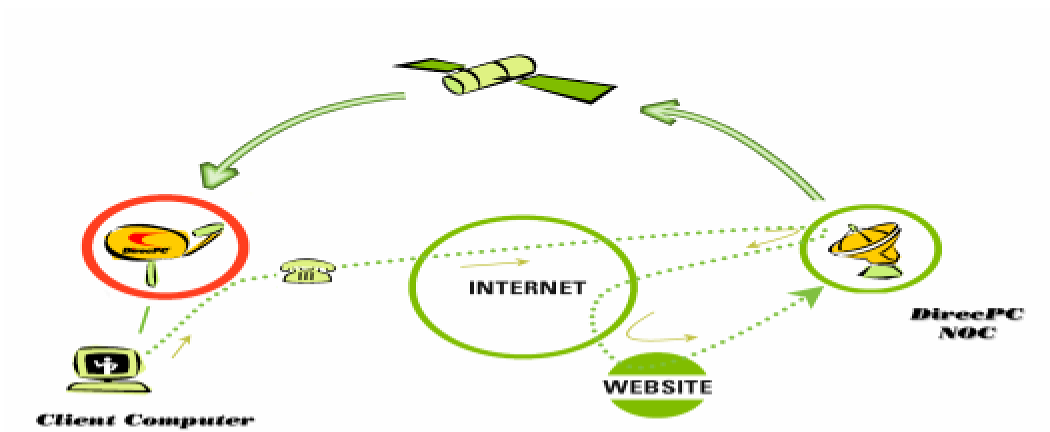




Star network (switch at hub) | Mesh network (switch at satellite)

Why do we choose this project?

- Commercial satellite companies (e.g., Loral, Hughes, Lockheed Martin) have announced plans to build large satellite systems to provide broadband data service.
- Our simulation may help improve TCP performance over long delay and error prone channels.



Project objective

- Implement an combination of four approaches to enhance the performance over satellite links coupling with the long delay and high Bit Error Rate characteristics.
  - Effect of window size RFC1323
  - Effect of Initial windows size RFC2581
  - Effect of Maximum segment size RFC2488
  - Comparison of different TCP algorithms RFC 2018
- Extend the authors' knowledge of TCP burst problem related to on board switch in GEO satellites.
  - TCP burst

Window size

- Large delay*bandwidth product

  W=B*RTT

  This product defines the amount of data a protocol should have " in flight".

Window size

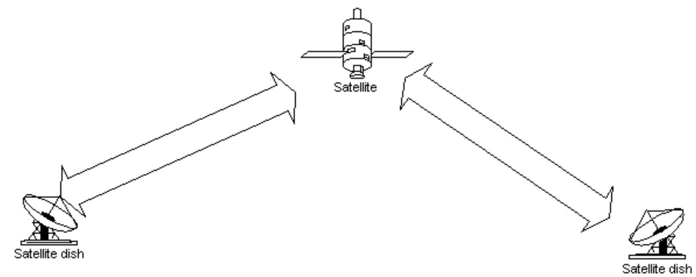
- The original TCP standard limits the advised window size by only assigning 16bits of header space for its value. (RFC793). Hence the advised window size can be no more than 64Kbytes.

$$\text{throughput} = \frac{CWND}{round\ trip\ time}$$

$$= \frac{64\ kbyte}{585\ ms} \approx 112,000\ bytes\ /\sec \approx 900\ kbps$$

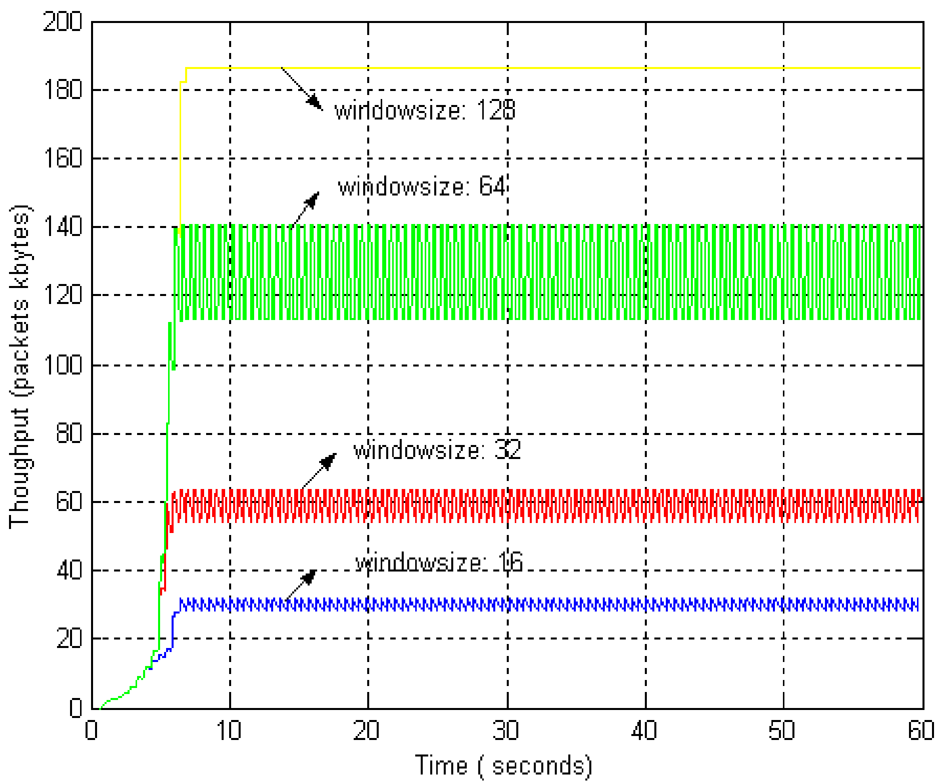
Window size

- Simulation scenario



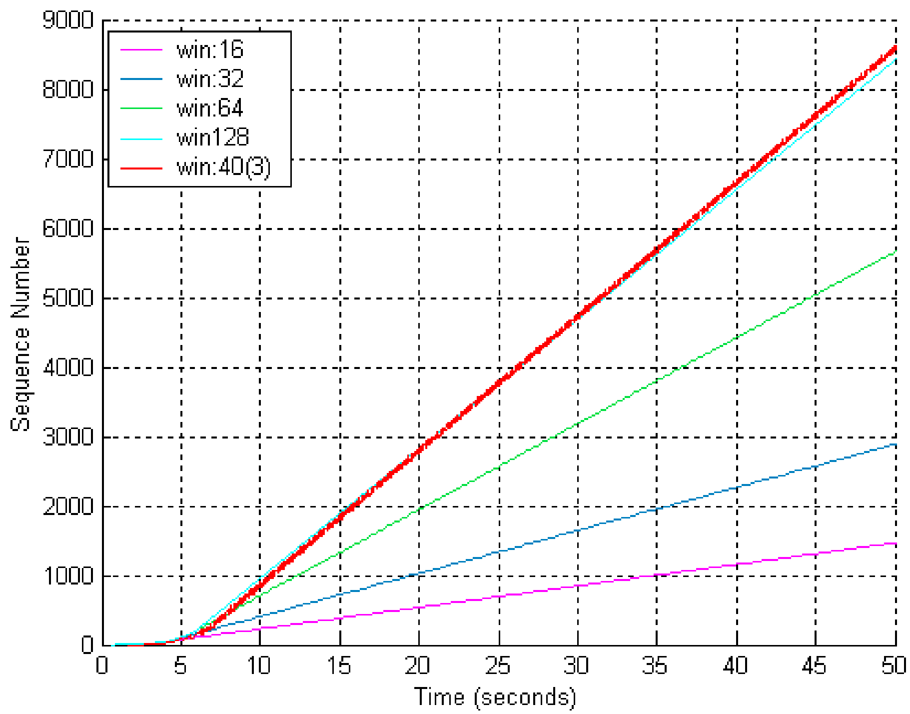
- Parameters:
  - choose various window size: 16, 32, 64, and 128.
  - T1 link: 1.544Mbps.
  - Set the receiver and sender buffer size greater than the delay bandwidth product, so that we can examine how window size affect on TCP throughput: 120.
  - Application: FTP

Window size (Result 1)



- **Brief Analysis**: Larger window size can help improve the throughput.

Window size (Result 2)



- Brief Analysis
  - Multiple long lived connections with small window size can still fully utilize the channel.
  - Three 40kbytes window connections can almost fully utilize the T1 channel.

Window size -Discussion

- Advantages:
  - It is ideal for the connections that transmit big files such as FTP application
- Disadvantages:
  - large window size can lead to more rapid use of the TCP sequence space.
  - large window size will also increase the multiple packets loss possibility.

What about short lived connection?

- Slow start is a safe guard against transmit inappropriate amount of data into the network.

- However, Slow start is particularly inefficient for short lived connection (Telnet) in large bandwidth-delay product network.

Initial congestion window size

- By increasing the initial CWND, more packets are sent during the first RTT.
- Trigger more ACKs, allowing congestion window to open more rapidly.

Maximum Segment Size

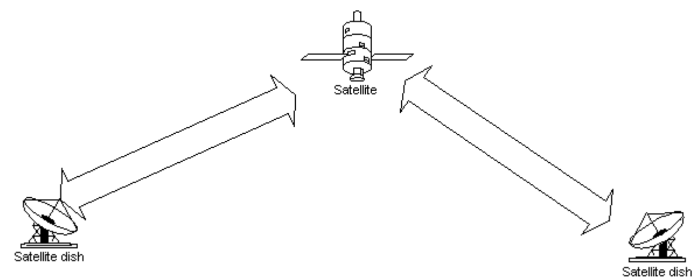
Maximum segment size (MSS):

MSS = MTU – TCP header – IP header

- MTU (or the maximum IP packet size): Maximum Transmission Unit

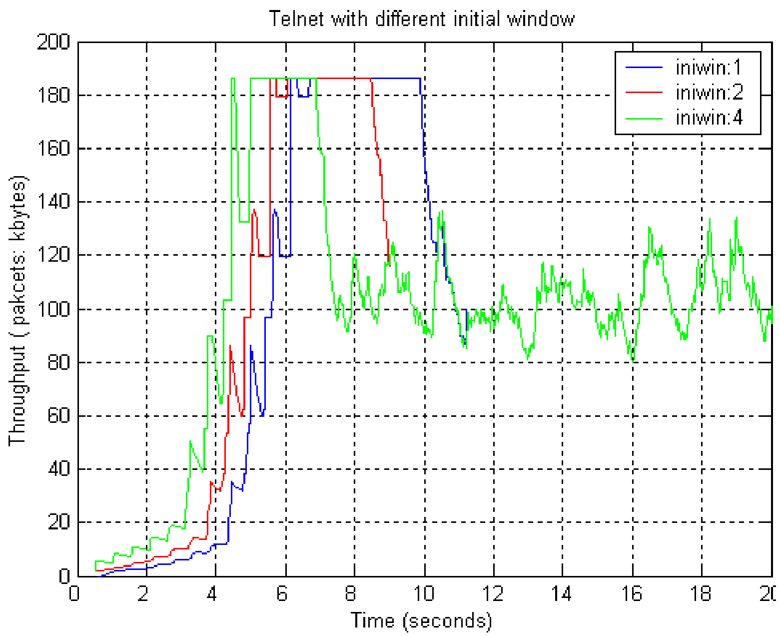
Initial CWND and MSS

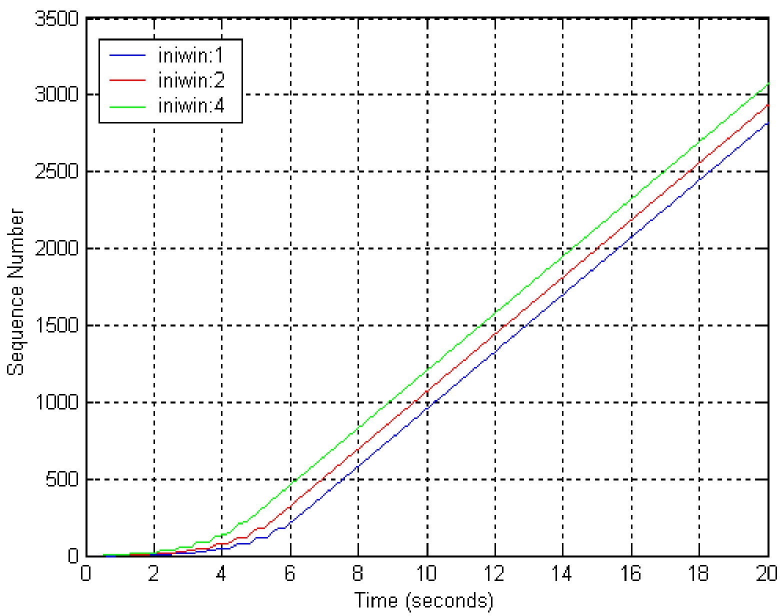
- Simulation scenario



- Parameters:
  - Set the initial window size: 1, 2, and 4.
  - Set packet size: 576, 1152, and 1728.
  - Set the advised window size: 128.
  - Application : telnet
  - Other parameters are default.

Initial window size (Result 1)

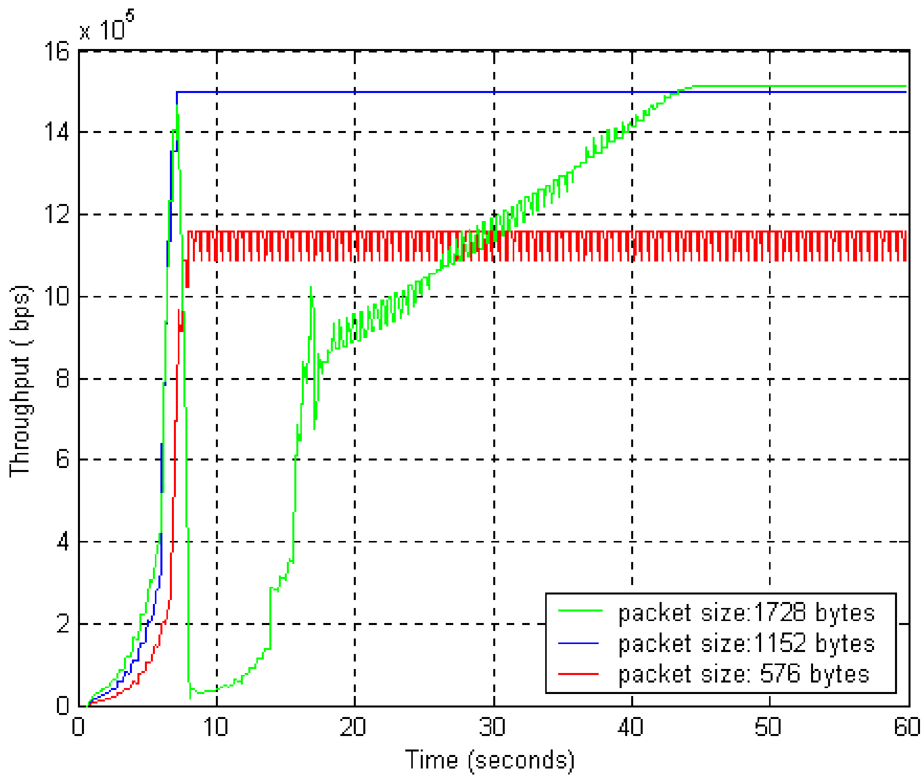




- Brief Analysis:

  Using larger initial window size can help reduce the slow start period, so allows sender send more packets at the same period of time.

Maximum Segment Size (Result)



- **Brief Analysis**:
  Choosing suitable maximum segment size can improve TCP throughput.

Initial CWND and MSS-Discussion

- Advantages:
  - It is ideal for the short lived connections such as Telnet application.
- Disadvantages:
  - Make traffic burst
  - Increase unnecessary drops for bigger packet size

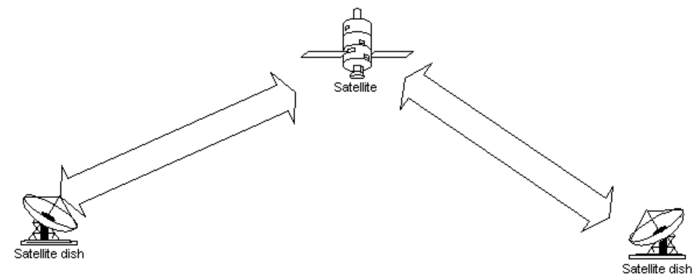
Comparison of different TCP algorithms

- New Challenge for TCP performance:
  - Large window size will also increase the multiple packets loss possibility.
  - Large MSS will easily be corrupted in wireless link.

- The need to evaluate the error correct ability of different TCP flavors.

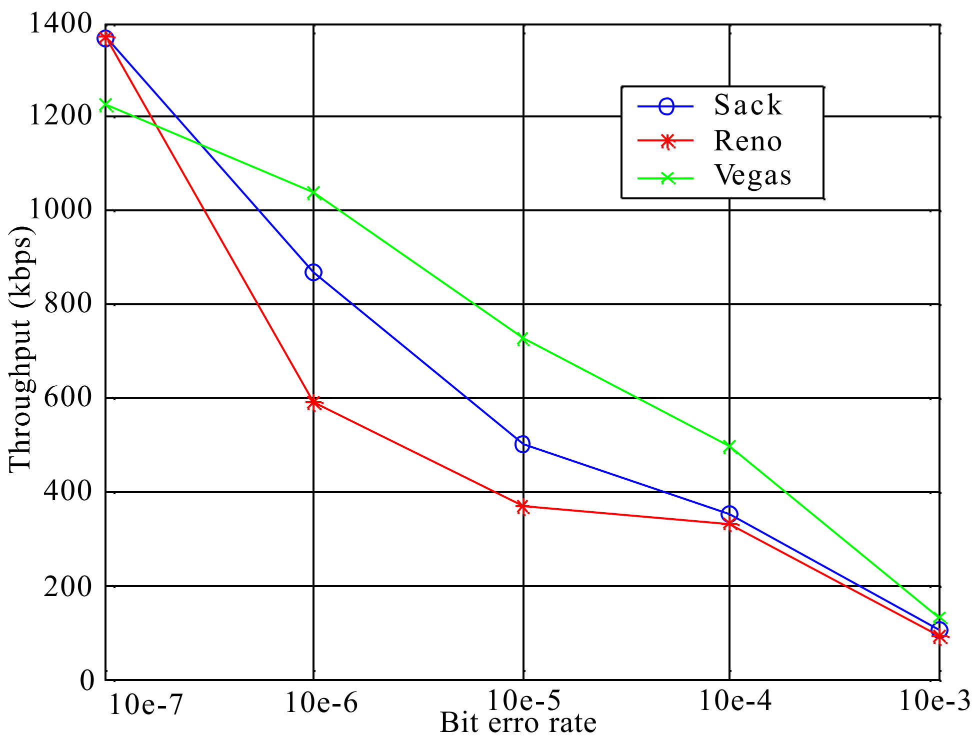
Comparison of different TCP algorithms

- Simulation scenario



- Parameters:
  - Add lose model, use different bit error rate : 10e-7, 10e-6, 10e-5, 10e-4, and 10e-3.
  - Compare TCP Sack with other TCP algorithms such as Reno and Vegas.

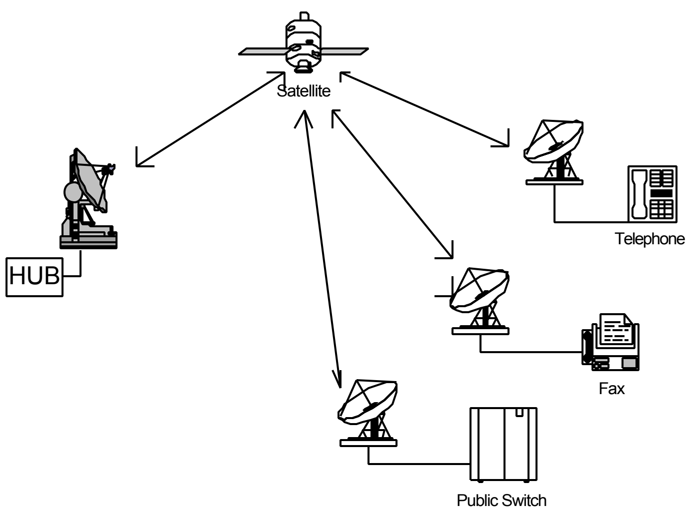
Result



- **Brief Analysis**:
  TCP Sack performs better than Reno. However, when bit error rate is from 5*10e-7 ~ 10e-3, Vegas is better than both Sack and Reno.

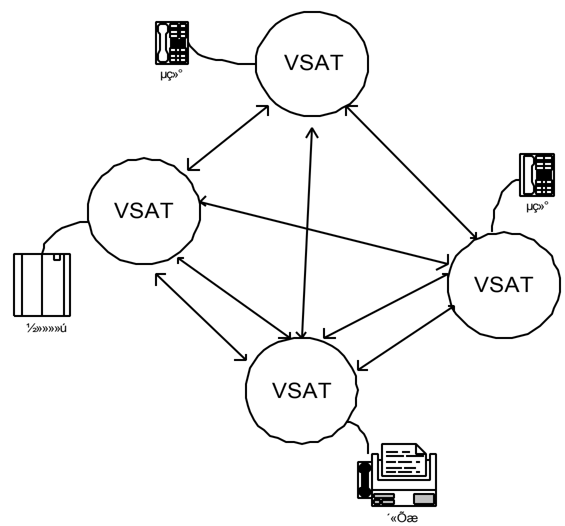
On board GEO satellite

Bent pipe GEO satellite



Star network (switch at hub)

On board switch GEO satellite



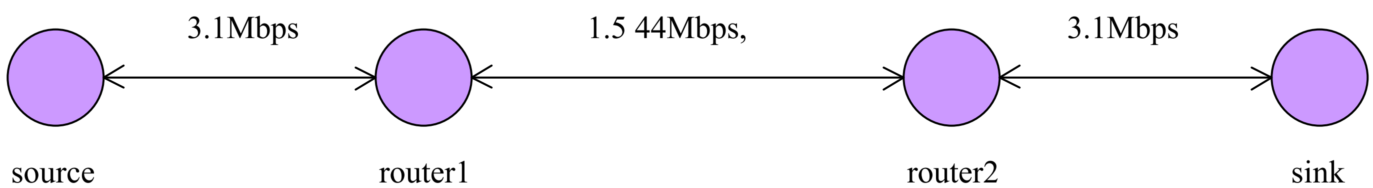
Mesh network (switch at satellite)

TCP Burst

- Larger window size and initial windows size will results in large burst.
- On board switch has limited buffer.
- New Challenge for TCP performance:
  - TCP burst Vs limited buffer size
  - Suitable buffer size

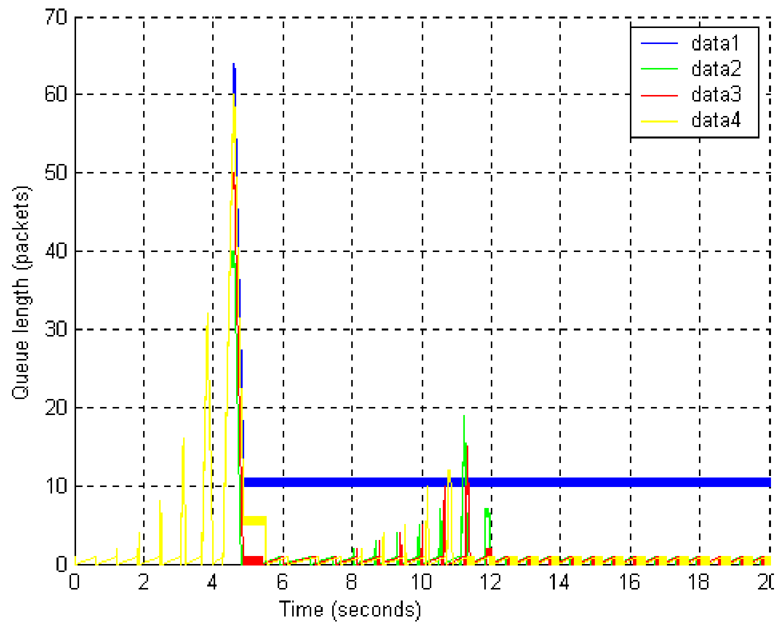
TCP Burst

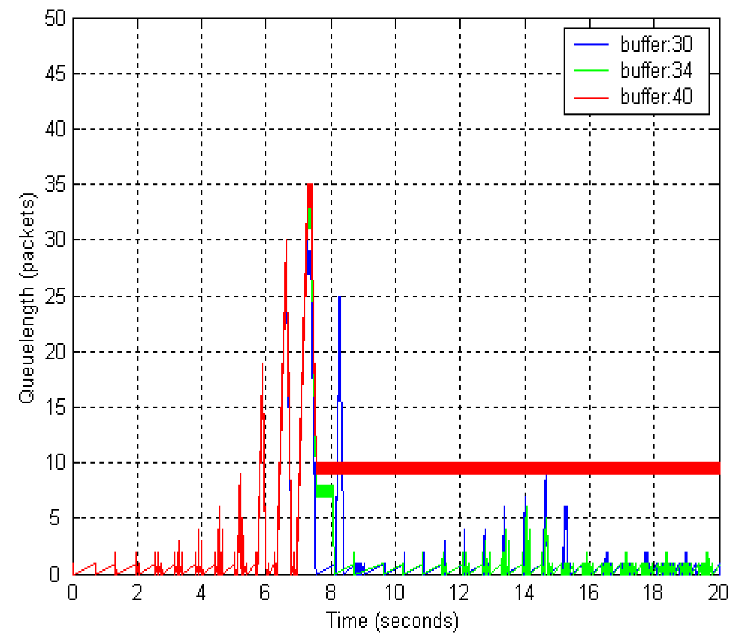
- Simulation scenario:



- Parameters:
  - Set various buffer size: 30, 34, 40, and 50.
  - Set the window size: 128.
  - T1 link : 1.544Mbps.
  - Other parameters are default.

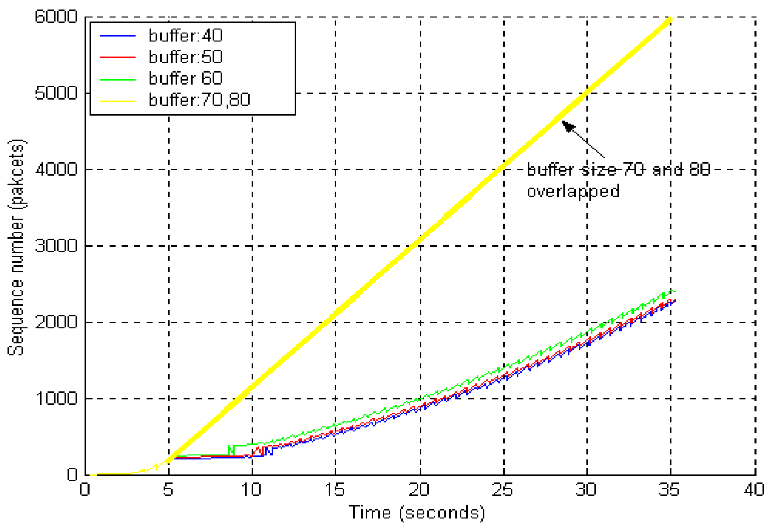
TCP Burst



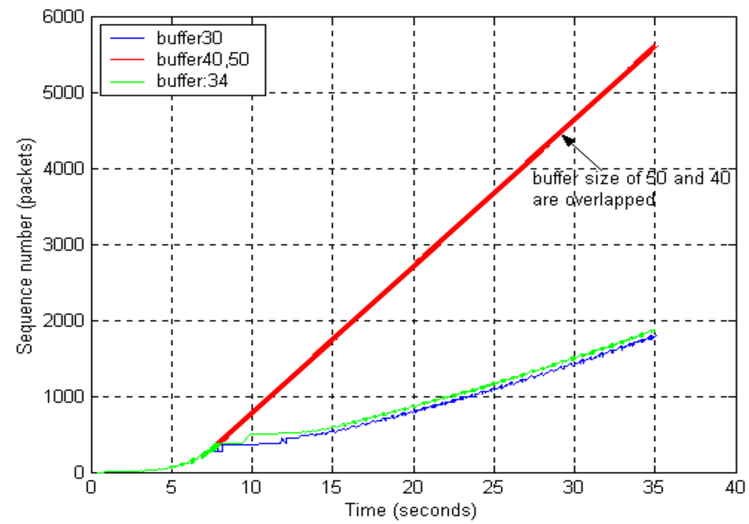


- Brief Analysis:
  - Increasing the buffer size will not alleviates the TCP burst
  - 2. After the slow start period, the queue occupancy decrease.

TCP Burst



with basic acknowledgment



with delayed acknowledgment

- Brief Analysis:
  Increasing the buffer size will not improve the TCP performance. If the buffer size is approximately greater than 1/2 of the window size when using basic ACK (accordingly, 1/3 when using delayed ACK), the throughput will reduce rapidly.

Conclusion

Experiment	Outcome
Window size	Larger window size can improve the performance
Initial window	Using a larger initial window can improve the throughput, especially for short transfers
Maximum segment size	Larger maximum segment size will improve the throughput, however, it may cause link congestion or router overload
Comparison of three TCP algorithms	TCP Sack is better than Reno under the error prone channel. However, Vegas performs even better than Sack if the bit error rate is large
TCP Burst	Burst has a severe influence on TCP performance

Conclusion

- For FTP application in satcom, implement SACK or Vegas with larger window size can greatly improve the TCP throughput.
- For Telnet application in satcom, it is ideal to implement SACK with larger initial window and MSS.
- The ideal buffer size of on board satellite router is ½ of the window size (standard ACK), 1/3 of the window size (delay ACK).

Reference

- [1] Van Jacobson. Congestion Avoidance and Control. In ACM SIGCOMM, 1988.
- [2] J. Mo, R. J. La, V. Anantharam, and J. Warland, ``Analysis and comparison of TCP Reno and Vegas," Proceedings of the Conference on Computer Communications (IEEE Infocom), New York, Mar. 1999.
- [3] L. Brakmo, S. O'Malley, and L. Peterson. TCP Vegas: New techniques for congestion detection and avoidance. In Proceedings of the SIGCOMM '94 Symposium (Aug. 1994) pages 24-35
- [4] K. Fall and S. Floyd, "Simulation-based comparisons of Tahoe, Reno, and SACK TCP," SIGCOMM Computer Communication Review, 26(3), July 1996
- [5] A Simulation Study of Paced TCP, Joanna Kulik, Robert Coulter, Dennis Rockwell, and Craig Partridge, September 1999
- [6] [RFC 2488] Enhancing TCP over Satellite Channels, Mailman, and D.Glover January 1999.
- [7] Larry L.Peterson and Bruce S.Davie. Computer networks: A system approach. Morgan Kaufman, 1996.
- [8] [RFC 2581] TCP Congestion Control, M. Allmanm, V. Paxson, W. Stevens, April 1999.
- [9] [RFC 1122] Transport Layer TCP
- [10] [RFC 1323] TCP Extensions for High Performance, V.Jacobson, R.Braden, D.Borman, 1992