A History of the Improvement of Internet Protocols Over Satellites Using ACTS

Mark Allman, NASA GRC/BBN

Hans Kruse, Shawn Ostermann, Ohio University

ACTS Conference 2000

May 2000

Motivation

- Delivering Internet content via satellite to places that are not necessarily well covered by good terrestrial connectivity.
- NASA is interested in possibly using off-the-shelf products for space communication.

Network Setup

- Half-to-full T1 ACTS channels
  - Some loop-back, some between GRC and OU
- $\approx$ 560–575 ms RTT
- Unless otherwise noted we used standard ACTS FEC
- NetBSD workstations as data clients and servers
- Cisco 25xx routers

TCP Problem 1

- The TCP *window size* (W) required to fill a network channel with BW bits/second of capacity and a round-trip time of RTT is:

$$W = BW \cdot RTT$$

- For a T1 ACTS circuit $W \approx 100\ KB$
- As originally written, TCP's maximum window size is 64 KB.
- So, TCP's maximum rate over an ACTS link is roughly 117 $KB/second$ regardless of the amount of capacity available.
  - For instance, an ACTS T1 circuit ($\approx 192\ KB/second$) can never be fully utilized.
  - (Note: This limit has been significantly raised since these experiments).

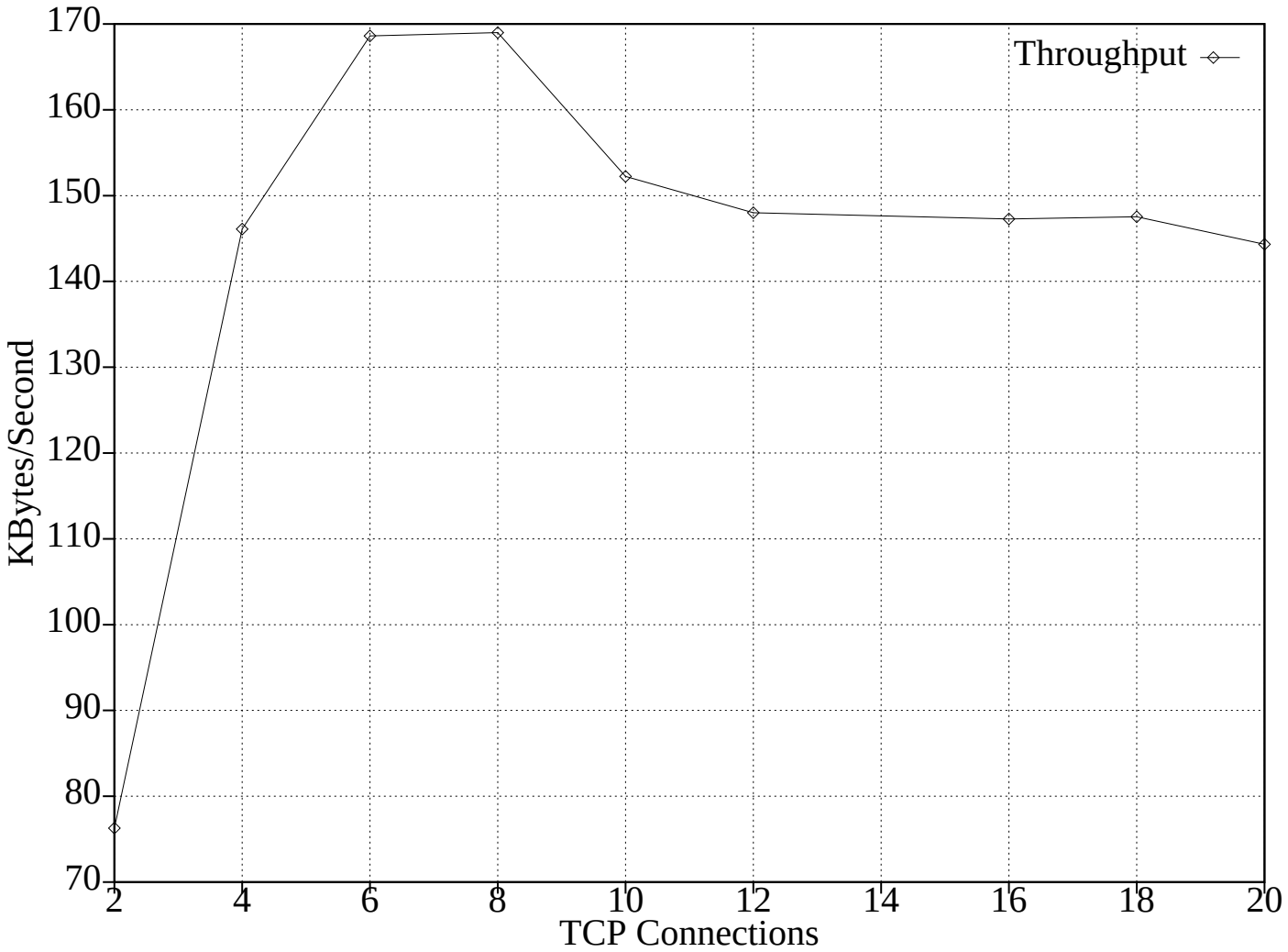
TCP Problem 2

- To avoid congestion collapse, a set of *congestion control* algorithms were added to TCP in 1988.
- The *slow start* is designed the gradually increase TCP's sending rate at the beginning of a transfer.
- Slow start works by sending a single segment into the network and waiting for the corresponding acknowledgment (ACK).
- In the remaining RTTs TCP doubles the number of segments sent per RTT, until...
  - There is no more data to send
  - TCP hits the maximum window size
  - TCP detects packet loss (i.e., congestion)

An Application Level Mitigation

- Our first cut at a "solution" to these problems was an application-layer modification to the FTP protocol.
- We designed an FTP client and server that would use multiple TCP connections to transfer a file, rather than the standard single connection.
  - This effectively increased TCP's maximum window size.

An Application Level Mitigation (cont.)



Standards-Based Solutions

- The IETF has come up with two mechanisms that help TCP over satellite channels:
  - RFC 1323 defines an option that allows TCP to use window sizes much larger than 64 KB.
  - RFC 2018 defines a selective acknowledgment (SACK) option that allows TCP to recover from lost segments more effectively.

Standards-Based Solutions (cont.)

- Congestion-free network:
  - TCP+Window Scaling performed nearly as well as *xftp* with 4 connections (i.e., full utilization for a long transfer)
- Congested network:
  - TCP+Window Scaling+SACK performed much better than TCP without SACK, but was outperformed by *xftp*.
    - *xftp* is more aggressive during congestion than standard TCP, so this result is understandable

Experimental Solutions

- Beginning slow start with an *initial congestion window* larger than 1 segment.
  - Our ACTS experiments show a 25% performance improvement when using a 4 segment initial congestion window to transfer a short file.
- Using *byte counting* rather than standard ACK counting to increase the congestion window.
  - Basing congestion window increase on the number of bytes acknowledged rather than the number of ACKs received makes the increase more accurate (due to delayed ACKs, ACK loss, etc.).
  - Our ACTS experiments show a 17% performance improvement when using byte counting.

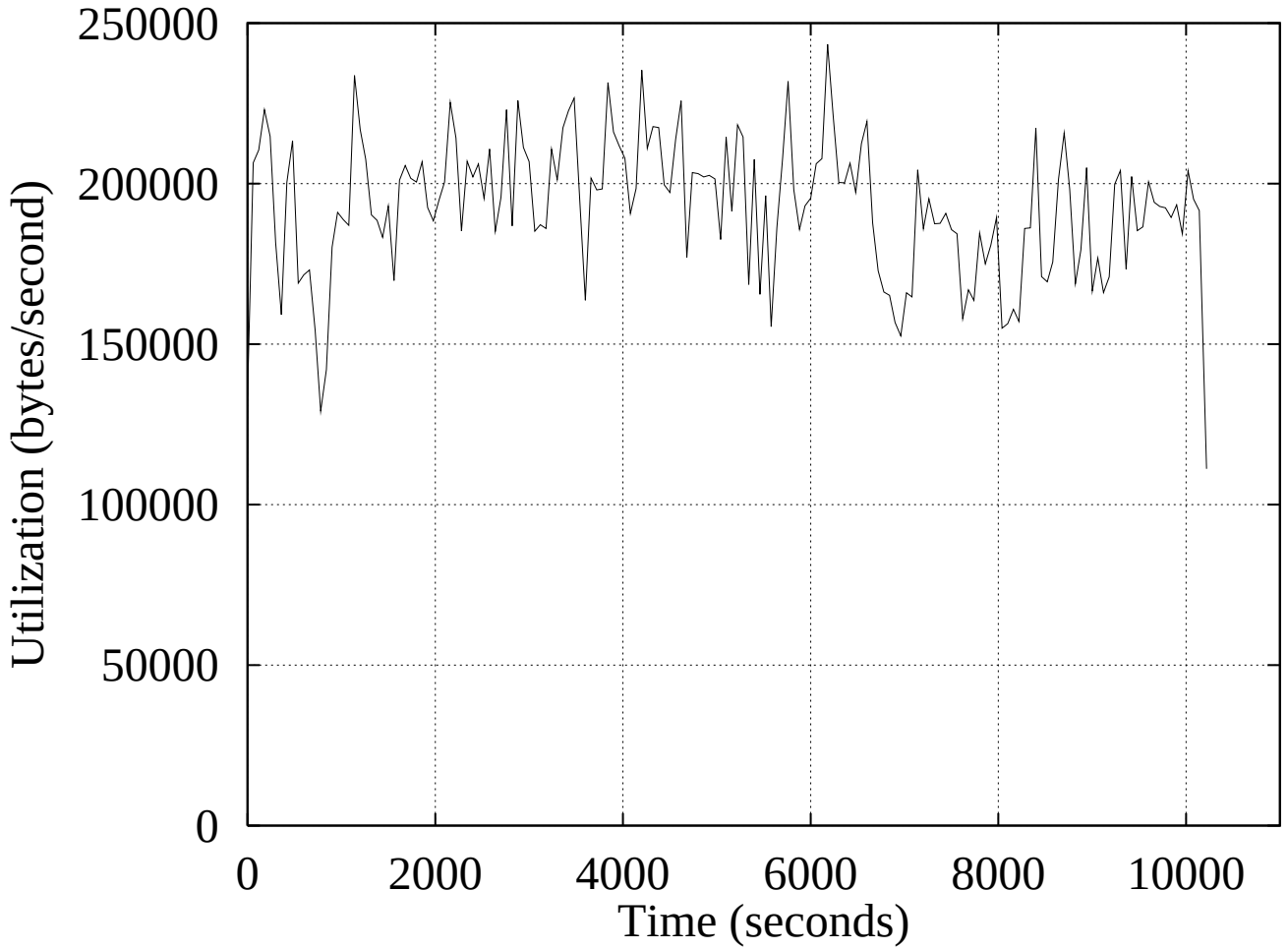
HTTP Experiments

- We used both HTTP/1.0 and HTTP/1.1 in our ACTS experiments, in conjunction with several options on both protocols.
- We found at least a factor of 2 difference in performance between the best set of options and the worst.
- Using a single HTTP/1.1 connection with the pipelining option provided the best performance.
- This set of experiments illustrates the importance of good design in application protocols and highlights the need to remain constantly vigilant as new application protocols are developed.

Representative Network Traffic

- Up to this point our experiments had consisted of only a handful of flows traversing the network simultaneously. But, this is not a realistic condition for production networks...
- Therefore, we wrote a tool that generates random network traffic that is based on network traffic observed in production networks.
  - Generates: WWW, FTP, SMTP, NNTP, Telnet
- We wanted to gauge how well a significant amount of network traffic could utilize a network path with a satellite channel.

Representative Network Traffic (cont.)



Impact of Bit-Errors

- As expected, a non-zero bit-error rate has the effect of reducing TCP performance because the segment losses are interpreted as indications of network congestion.
  - TCP reduces the sending rate when detecting network congestion.
- A more verbose discussion of our ACTS tests and results will be given on Friday morning.

Standards Contributions

- These IETF RFCs were directly or indirectly influenced by our ACTS experiments:
  - *RFC 2414*: Experimental proposal to increase the initial congestion window size.
  - *RFC 2488*: Discussion of the standard mechanisms that should be implemented when using TCP over satellite channels.
  - *RFC 2581*: Standardized the use of a 2 segment initial congestion window and byte counting during congestion avoidance.
  - *RFC 2760*: Outline of ongoing research in TCP over satellite networks.

Conclusions

- TCP *can* use the full capacity of a satellite channel when transferring large amounts of data.
- Short transfers often underutilize the capacity.
  - We have mitigated this, but future research is needed.
- Application layer protocols can have a big impact on performance. We must be vigilant when we design these protocols.
- A realistic mix of network traffic *can* fully utilize the available capacity of a satellite channel.
- Future work (starting tomorrow!) includes investigating additional host and router mechanisms to further increase data transmission performance over satellite links.

Acknowledgments

Our work simply would not have been possible without the assistance, patience and hard work of many people in the ACTS operations team and the research community. Our thanks to all!