Advanced Computer Networks

Congestion Control over Large Bandwidth-Delay Product Networks

Jianping Pan
Summer 2007

Review: TCP congestion control

- Loss-based
  - how to detect/react to packet losses
- Delay-based
  - how to react to delay changes
- Rate-based
  - how to determine the TCP throughput
- AIMD-based
  - how to choose AIMD parameters to be TCP friendly

New challenges

- Large bandwidth-delay product networks
  - aka “long-fat” (elephant) networks
  - example by Floyd: “A standard TCP connection with
    - 1500-byte packets,
    - a 100ms round-trip time, and
    - a steady-state throughput of 10Gbps,
  - would require
    - an average congestion window of 83,000 packets and
    - at most one drop (mark) every 5,000,000,000 packets (or equivalently, at most one drop every 1 2/3 hours).
  - This is *not* realistic”

Q: 1 in 5 billion packets?

Another example

- Scenarios
  - 10 Gbps point-to-point, dedicated link
  - 1500-byte packets
  - 100 ms round-trip time
  - large enough sender and receiver buffer
- Questions
  - how long does it take to fill the pipe initially?
  - after the first timeout?
  - after the follow-on triple dupack?
  - what is the link utilization?

TCP congestion control

- AI
  - on a new ack
  - cwnd = cwnd + MSS*MSS/cwnd
  - equivalently, cwnd += MSS for every RTT
    - or cwnd += MSS/b if acknowledging every b packets
- MD
  - on a loss event
  - cwnd = cwnd/2
  - AI follows if Fast Recovery
    - cwnd/2 RTT to increase from cwnd/2 to cwnd

Critics on TCP congestion control

- Congestion loss vs transmission error
  - e.g., wireless links
  - approaches: TCP over wireless
    - transport-layer approaches
    - link-layer approaches
    - hybrid approaches
- "(1, 0.5)-AIMD is too conservative/aggressive"
  - Discussion
    - when is (1, 0.5)-AIMD good?
    - when is not?

Other issues with “elephant” networks

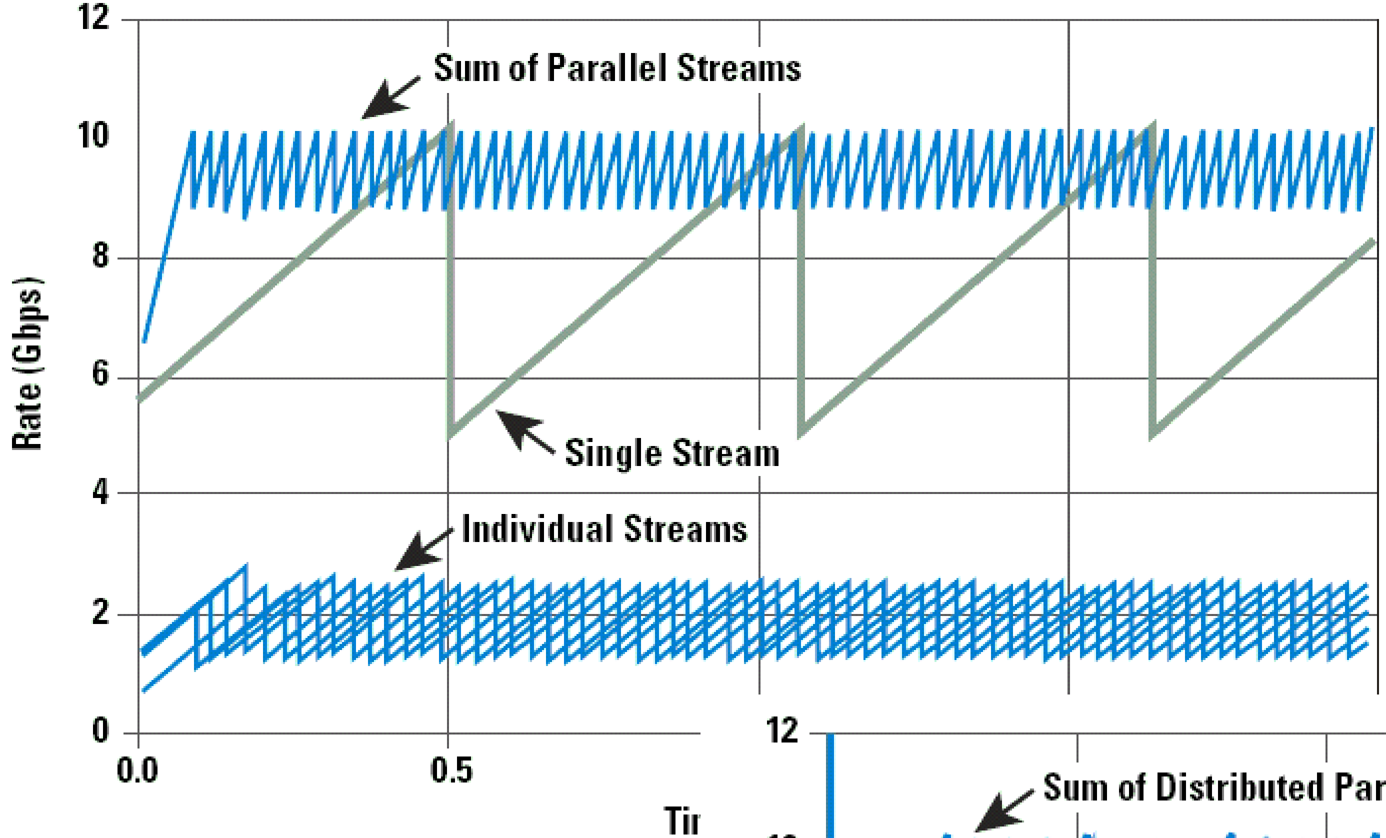
- Window size
  - TCP: 16-bit window size; byte sequence
    - i.e., 64 KB unacknowledged data at most
  - on high-speed links
    - transmission time << propagation time < round-trip time
- Sequence space
  - TCP: 32-bit sequence space; byte sequence
- Approach
  - TCP window scale option
    - left-shift at most 14 bits
    - i.e., 1 GB

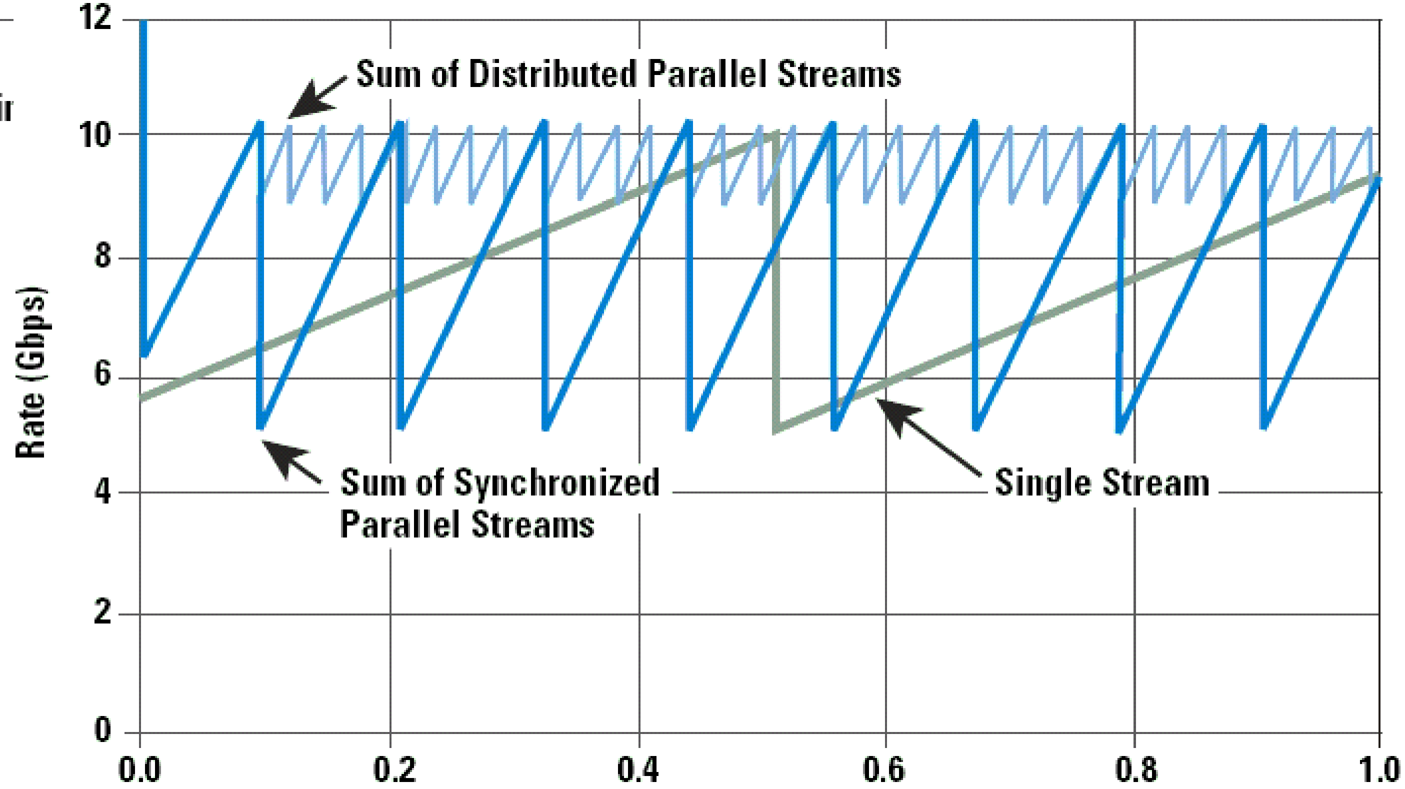
Q: why “at most 14 bits”?

Approaches

- Multi-TCP
  - multiple TCP connections
    - between the same pair of endpoints, or
    - from many endpoints to one endpoint (data sink)
  - good: no changes to TCP
  - bad: many TCP connections in one endpoint
    - appropriate data splitting and reassembly
  - ugly: synchronization between connections
- Newer TCP
  - goal: work well in elephant networks
  - also work well with legacy TCP in regular networks

Multi-TCP



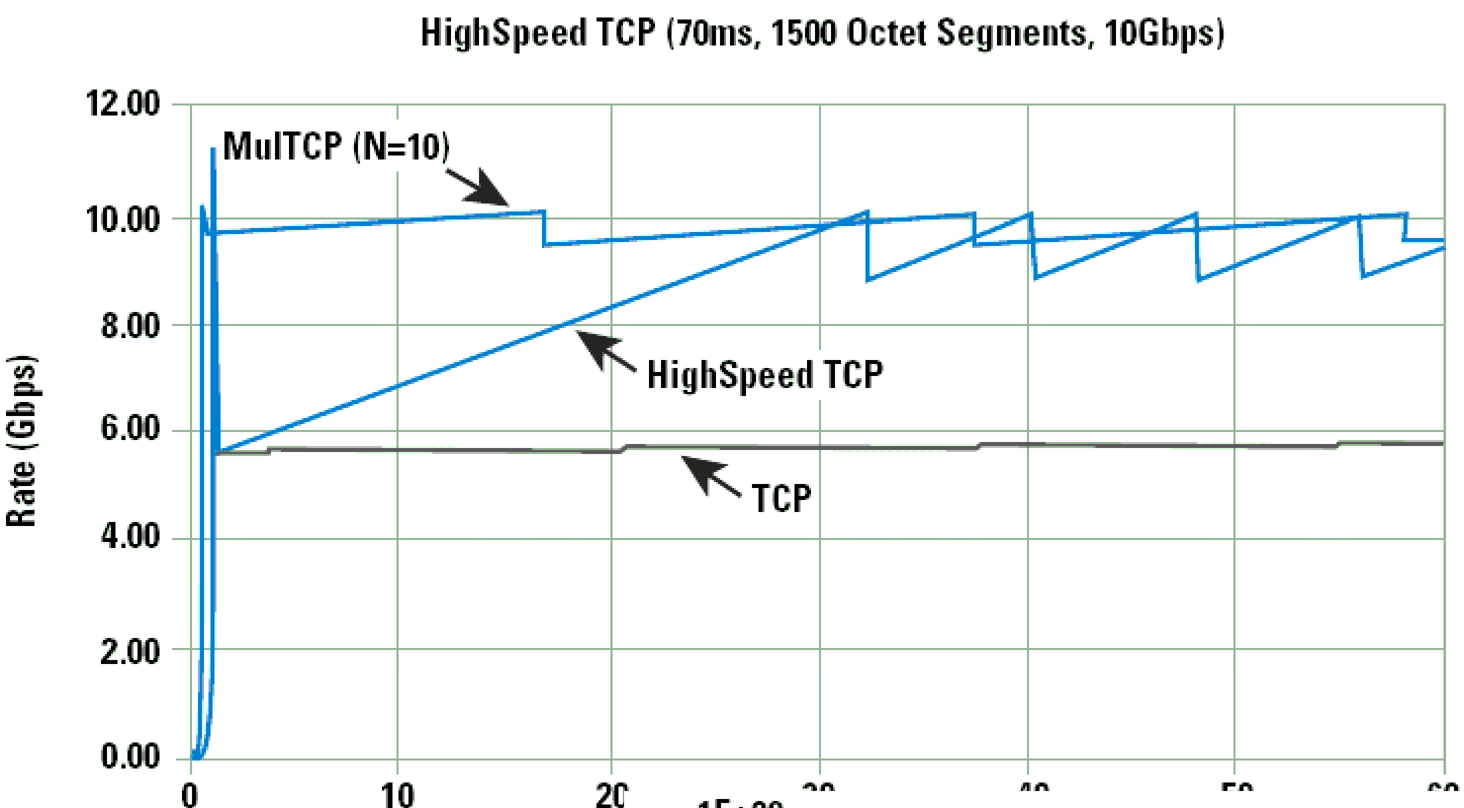


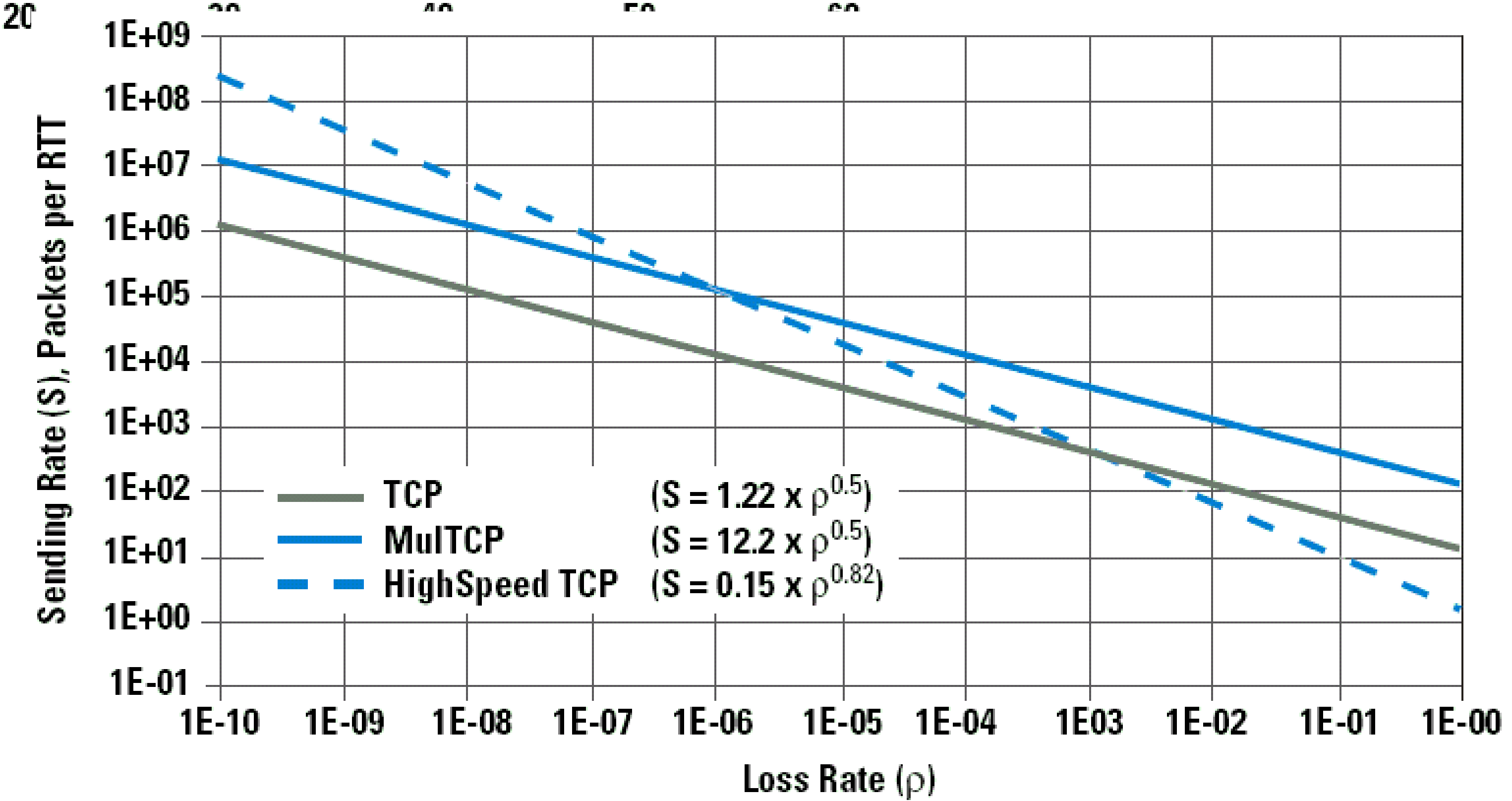
http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_9-2/gigabit_tcp.html

High-Speed TCP

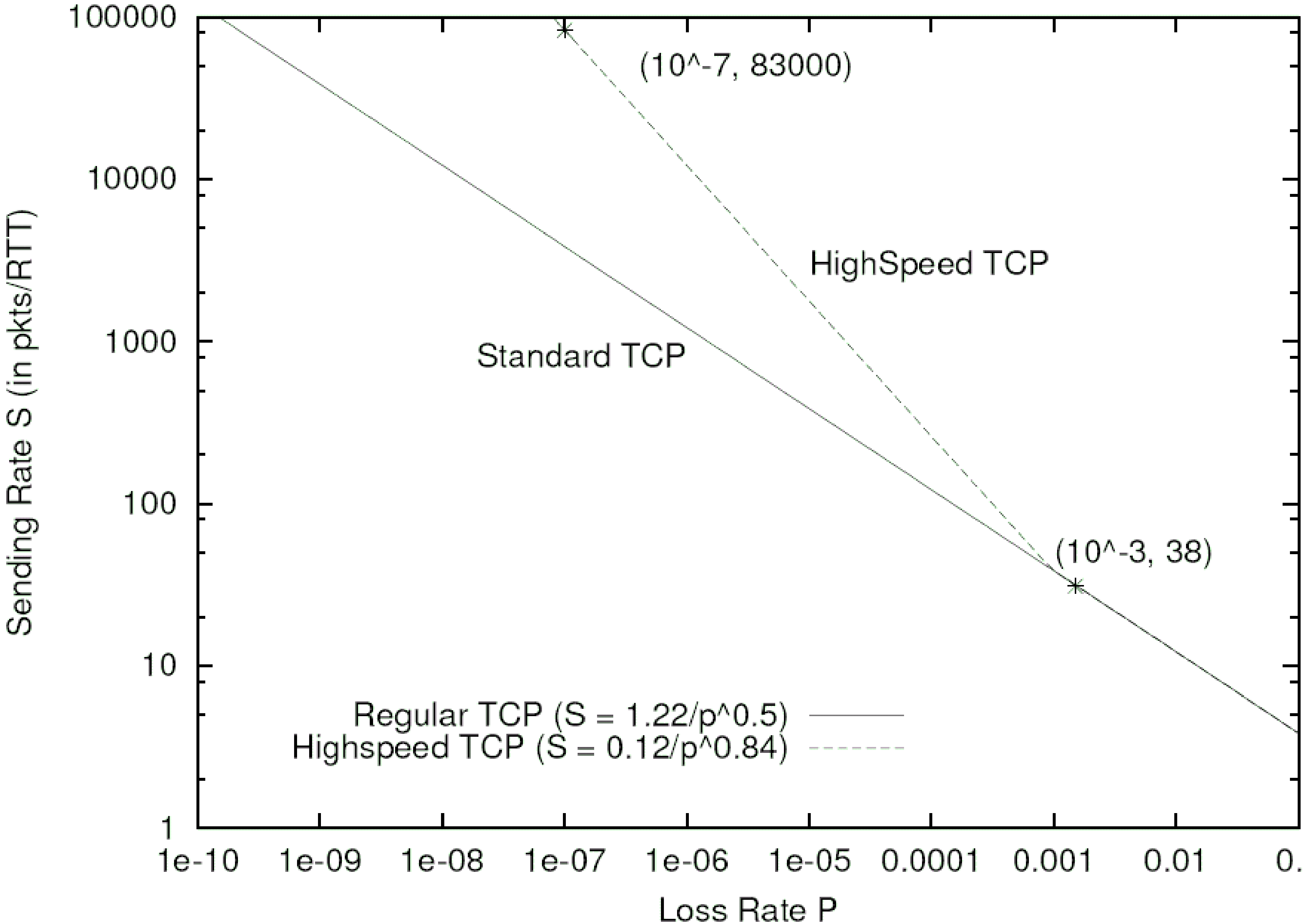
- MuTCP: to emulate N TCP connections
  - AI: increase by N*MSS per RTT
  - MD: reduce by 1/(2N) per loss event
  - not TCP-friendly even in non-elephant networks
- HS-TCP by Floyd
  - AI: increase by a(cwnd) per RTT
    - a(cwnd): a function of cwnd
    - higher cwnd, larger a(cwnd)
  - MD: reduce by b(cwnd) per RTT
    - higher cwnd, smaller b(cwnd)
  - can maintain TCP-friendly in non-elephant networks

MuTCP, HS-TCP


HighSpeed TCP (70ms, 1500 Octet Segments, 10Gbps)
MulTCP (N=10)
HighSpeed TCP
TCP
Rate (Gbps)
12.00
10.00
8.00
6.00
4.00
2.00
0.00
0
10
20

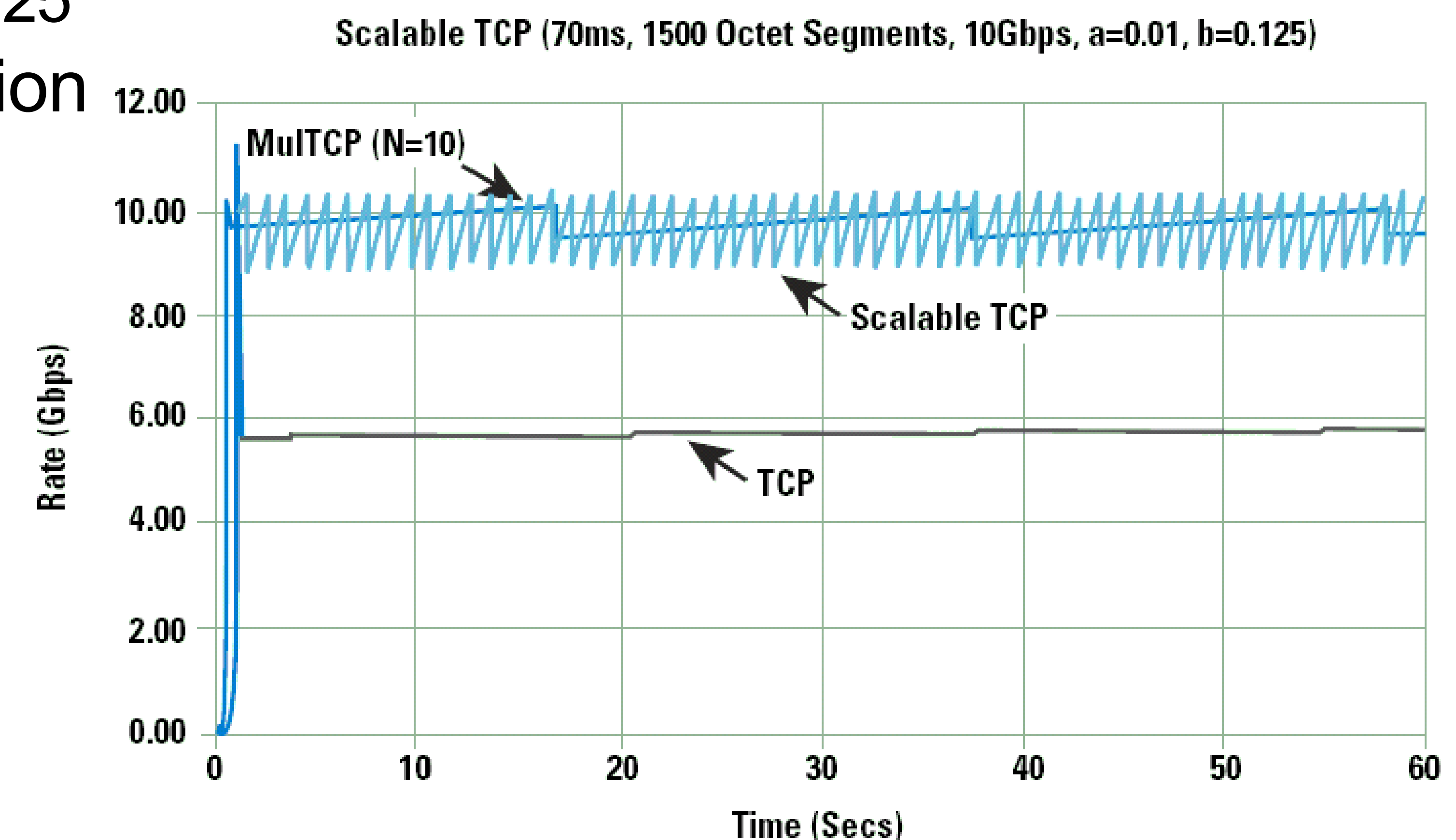

Sending Rate (S), Packets per RTT
1E+09
1E+08
1E+07
1E+06
1E+05
1E+04
1E+03
1E+02
1E+01
1E+00
1E-01
TCP (S = 1.22 x ρ^0.5)
MulTCP (S = 12.2 x ρ^0.5)
HighSpeed TCP (S = 0.15 x ρ^0.82)
1E-10
1E-09
1E-08
1E-07
1E-06
1E-05
1E-04
1E03
1E-02
1E-01
1E-00
Loss Rate (ρ)

TCP-friendly HS-TCP



Scalable TCP

- S-TCP by Kelly
  - MI: increase by a on each new ack
    - multiplicative increase every RTT; e.g., a = 0.01 MSS
  - MD: decrease by b on every loss event
    - e.g., b = 0.125
  - more oscillation

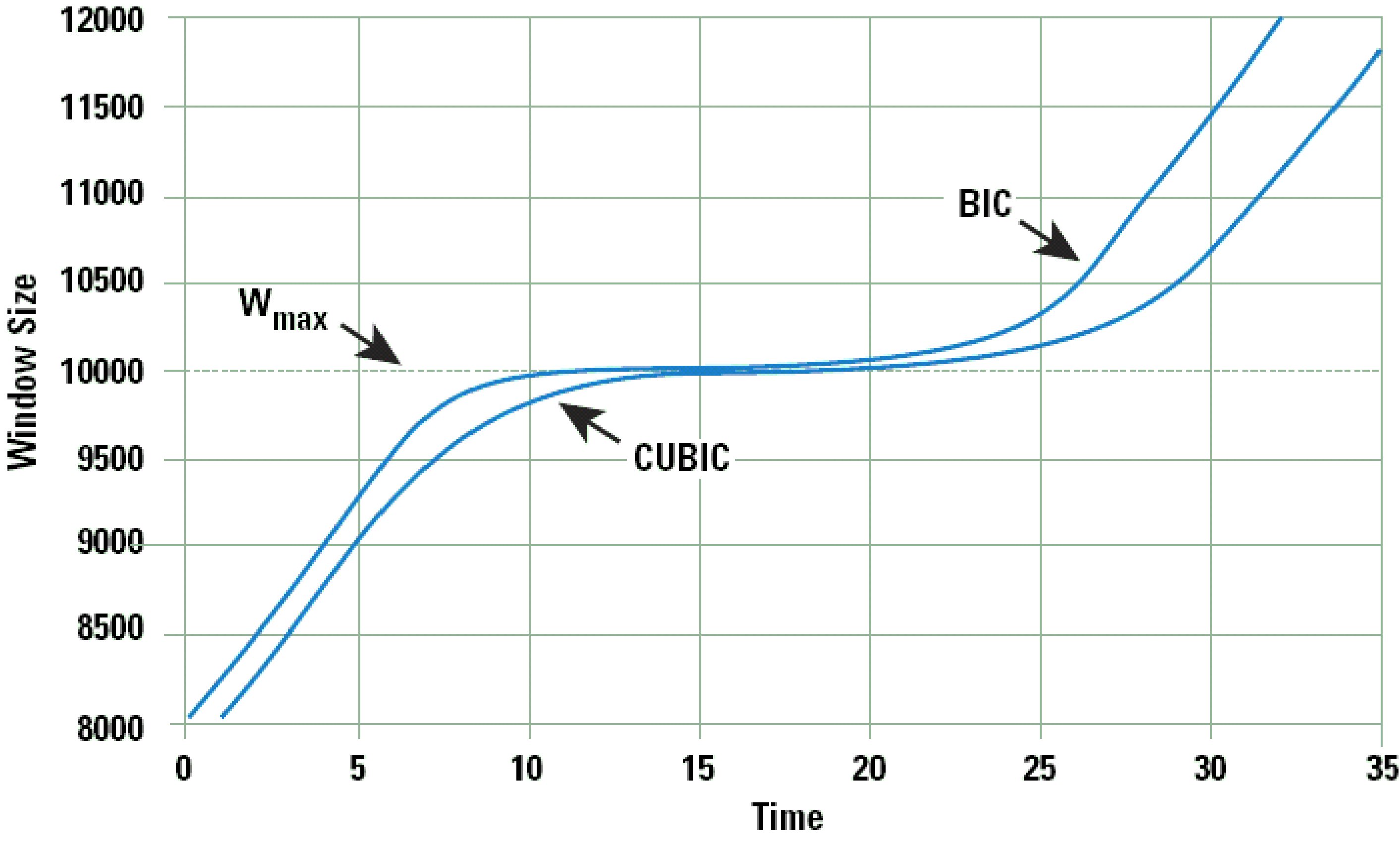


Fast AQM Scalable TCP

- FAST TCP by Low
  - built upon TCP Vegas
    - delay-based congestion control
  - slower than slow-start
    - adjust cwnd every other RTT
    - exit when achievable throughput is lagging behind more a threshold, rather than packet loss
  - multiplicative increase
    - when below equilibrium, approach faster
  - exponential convergence
    - move half-way between the current and target value

BIC and CUBIC

- BIC: binary increase congestion control
  - reduce cwnd on loss event
  - remember cwnd before loss event
  - binary search between current and last cwnd during congestion avoidance
- CUBIC
  - 3rd-order polynomial function
  - better stability



Student Presentation

- Emad Shihab: XCP
  - [KDR02] Dina Katabi, Mark Handley, and Chalrie Rohrs. Congestion Control for High Bandwidth-Delay Product Networks. In the proceedings on ACM Sigcomm 2002. [XCP]

Further discussion

- TCP congestion control
  - a long-thriving research thrust
- Network protocols are essentially driven by
  - communication technologies
  - application requirements
  - they often change!

Type	Control Method	Trigger	Response
TCP	AIMD(1,0.5)	ACK response Loss response	$W = W + 1/W$ $W = W - W \times 0.5$
MulTCP	AIMD(N,1/2N)	ACK response Loss response	$W = W + N/W$ $W = W - W \times 1/2N$
HighSpeed TCP	AIMD(a(w), b(w))	ACK response Loss response	$W = W + a(W)/W$ $W = W - W \times b(W)$
Scalable TCP	MIMD(1/100, 1/8)	ACK response Loss response	$W = W + 1/100$ $W = W - W \times 1/8$
FAST	RTT Variation	RTT	$W = W \times (\text{base RTT}/\text{RTT}) + \alpha$

This lecture

- TCP over “long-fat” networks
  - problems and approaches
  - schemes
    - HSTCP, Scalable TCP, FAST
    - XCP
- Explore further
  - Internet Congestion Control Research Group
  - Internet2 Land Speed Record (LSR) http://www.internet2.edu/lsr/
  - Supercomputing Bandwidth Challenge (BWC)

Next lectures

- A new chapter
  - network routing
- Required reading
  - [KZ90] A. Khanna and J. Zinky, "A Revised ARPANET Routing Metric," ACM SIGCOMM '89, pp. 45-56, September 1989.
  - [LMJ97] C. Labovitz, G. R. Malan, and F. Jahanian, "Internet Routing Instability". In Proceedings of ACM SIGCOMM'97, September 1997.
  - [GR00] Lixin Gao and Jennifer Rexford, "Stable Internet Routing Without Global Coordination". In Proceedings of the 2000 ACM SIGMETRICS international conference on Measurement and modeling of computer systems. 2000.