TCP Performance

- Effective over a wide range of capacities
- A lot of operational experience

- Periodic loss (macroscopic) model shows that throughput is inversely proportional to
  - square root of loss probability *p*
  - RTT
  - average sending rate = sqrt(1.5/*p*) / RTT

TCP Futures: TCP over "long, fat pipes"

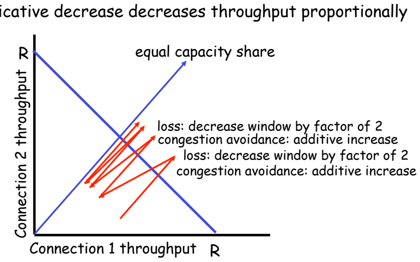
- Example: 10,000-bit segments, 100ms RTT, want 10 Gbps throughput
- Requires window size W = 100,000 in-flight segments
- Throughput in terms of loss rate: $\frac{1.22}{RTT\sqrt{p}}$

- p = 1.5 x 10^{-10} *Wow*
- New versions of TCP for high-speed

Why is TCP fair?

Two competing sessions (with same RTT):

- Additive increase gives slope of 1, as throughout increases
- multiplicative decrease decreases throughput proportionally



Delay modeling

Q: How long does it take to receive an object from a Web server after sending a request?

Ignoring congestion, delay is influenced by:

- TCP connection establishment
- data transmission delay
- slow start

Notation, assumptions:

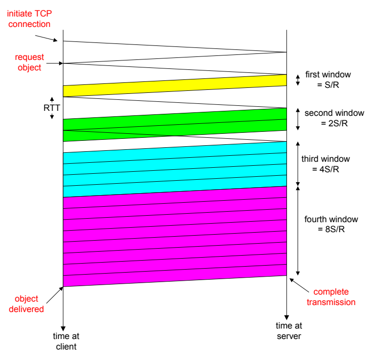
- Assume one link between client and server of rate C
- S: MSS (bits)
- O: object size (bits)
- no retransmissions (no loss, no corruption)

TCP Delay Modeling: Slow Start

Delay components:
- 2 RTT for connection estab and request
- O/C to transmit object

Example:
- O/S = 15 segments

Server idles 2 times due to slow start



Chapter 3: Summary

- principles behind transport layer services:
  - multiplexing, demultiplexing
  - reliable data transfer
  - flow control
  - congestion control
- instantiation and implementation in the Internet
  - UDP
  - TCP

Next:
- leaving the network "edge" (application, transport layers)
- into the network "core"