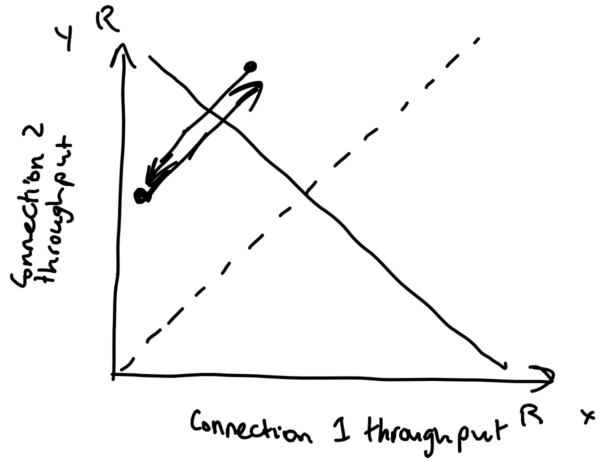
Kyle Romero

Problem 1

Suppose that instead of a multiplicative decrease, TCP decreased the window size by a constant amount. Would the resulting AIAD algorithm converge to an equal share algorithm? Justify your answer using a graphical diagram similar to Slide 100 of the lecture.

Write your solution to Problem 1 in this box

AIAD does not converge to an equal share algorithm. In order to show that this the case, we should observe the following example.



It should be noted that in the AIAD example above, the window sizes only move along a line with a slope of 1. Therefore, the algorithm never converges to an equal share algorithm. Equal share is denoted by the dotted line along $y = x$.

Problem 2

TCP is a very symmetric protocol, but the client/server model is not. Consider an asymmetric TCP-like protocol in which only the server side is assigned a port number visible to the application layers. Client-side sockets would simply be abstractions that can be connected to server ports. Can you propose header data and connection semantics to support this. What will you use to replace the client port number?

Write your solution to Problem 2 in this box

If we got rid of the port number, we would still need some kind of way to demultiplex traffic being received by the client. Therefore, I would propose having a connection number that corresponds to a client-side socket. This would introduce no new header fields and only substitute the port # field for a connection #. This asymmetric TCP-like protocol would require slighty different socket objects for clients and servers. The server-side socket object would need to be binded to a port # at creation while a client-side socket object would not.

Problem 3

On the TCP throughput, in the period of time from when the connections rate varies from W/(2 RTT) to W/RTT, only one packet is lost (at the very end of the period).

(a) Show that the loss rate (fraction of packets lost) is equal to L=lossrate= $1/(3/8W^2 + 3/4W)$

(b) Use the result above to show that if a connection has loss rate L, then its average rate is approximately given by $\simeq 1.22 \times MSS/(RTT \times \sqrt{L})$

Write your solution to Problem 3 in this box

a) $\text{lossrate} = \dfrac{\#\text{ of packets lost}}{\#\text{ of packets sent}}$

$$\#\text{ of packets sent} = \sum_{n=0}^{\frac{w}{2}} \left(\frac{w}{2} + n\right) = \sum_{n=0}^{w/2} \left(\frac{w}{2}\right) + \sum_{n=0}^{\frac{w}{2}} (n)$$

$$= \left(\frac{w}{2}+1\right)\left(\frac{w}{2}\right) + \frac{\left(\frac{w}{2}\right)\left(\frac{w}{2}+1\right)}{2} = \frac{w^2}{4} + \frac{w}{2} + \frac{w^2}{8} + \frac{w}{4}$$

$$= \frac{3w^2}{8} + \frac{3w}{4}$$

$\#$ of packets lost $= 1$, for one cycle of above $\#$ of packets

so, loss rate $= \dfrac{1}{\frac{3w^2}{8} + \frac{3w}{4}}$

b) If we have $\frac{3w^2}{8} >> \frac{3w}{4}$, then $L \approx \frac{8}{3w^2}$ and $w \approx \sqrt{\frac{8}{3L}}$

$$\text{avg throughput} = \frac{3}{4} w \cdot \frac{MSS}{RTT} = \frac{3}{4}\left(\sqrt{\frac{8}{3L}}\right) \cdot \frac{MSS}{RTT}$$

$$= \frac{1.22 \cdot MSS}{RTT \cdot \sqrt{L}}$$

Problem 4

You are designing a reliable, sliding window, byte-stream protocol similar to TCP. It will be used for communication with a geosynchronous satellite network, for which the bandwidth is 1 Gbps and the RTT is 300 ms. Assume the maximum segment lifetime is 30 seconds.

(a) How many bits wide should you make the ReceiveWindow and SequenceNum fields? (ReceiveWindow is also called "Advertised Window" in some other textbooks.)

(b) If ReceiveWindow is 16 bits, what upper bound would that impose on the effective bandwidth?

Write your solution to Problem 4 in this box

a) The ReceiveWindow needs to be large enough for the connection between the sender and satellite to be full. This would correspond to supporting (1 Gbps)(300 ms) = .3 Gb. We will need at least 26 bits to address this # of Gb.

The SequenceNum field needs to be large enough that it does not wrap around before an original segment is removed from pipe - this unit of time is called a maximum segment lifetime. So,

(30 seconds)(1 Gbps) = 30 Gb / max segment lifetime

= 3.75 x 10^9 bytes / max segment lifetime

Since the seq # could address any of these bytes, we would need at least 32 bits to be able to request each of these bytes.

b) If receive Window were full, we would limit the amount of data we could transfer to:

2^{16} / .300 seconds = 218453 bit/sec = 218.5 Kb/s

Problem 5

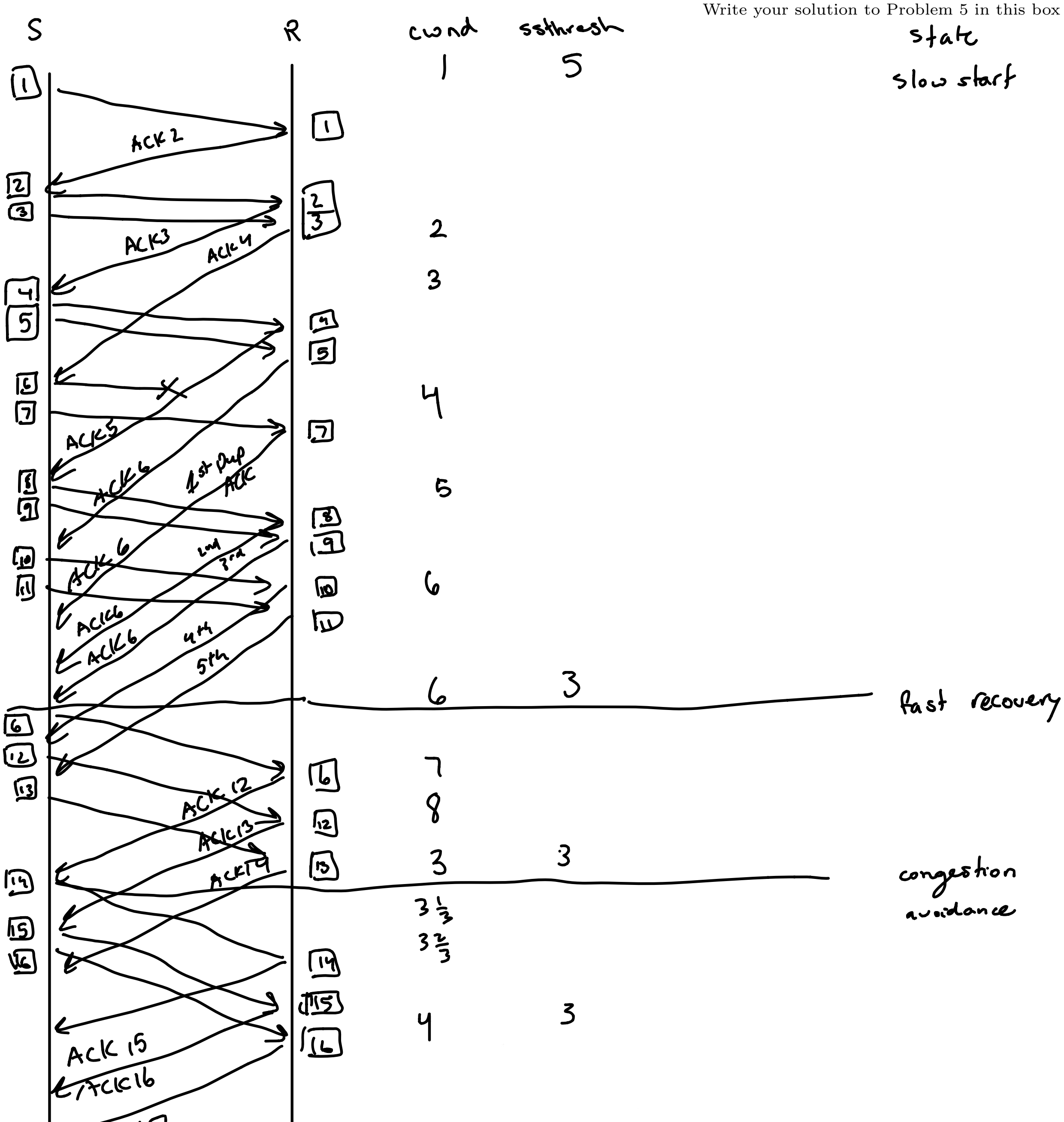
Consider the evolution of a TCP connection with the following characteristics. Assume that all the following algorithms are implemented in TCP congestion control: slow start, congestions avoidance, fast retransmit and fast recovery, and retransmission upon timeout. If ssthresh equals to cwnd, use the slow start algorithm in your calculation.

- The TCP receiver acknowledges every segment, and the sender always has data segments available for transmission.
- The network latency in sending a segment (header and payload) from the sender to the receiver is 30ms and the network latency in sending an acknowledgment (header only) from the receiver to the sender is 20ms. Ignore packet-processing delays at the sender and the receiver.
- Initially ssthresh at the sender is set to 5. Assume cwnd and ssthresh are measured in segments, and the transmission time for each segment is negligible.
- Retransmission timeout (RTO) is initially set to 500ms at the sender and is unchanged during the connection lifetime. The RTT is 100ms for all transmissions.
- The connection starts to transmit data at time t = 0, and the initial sequence number starts from 1. TCP segment with sequence number 6 is lost once (i.e., it sees segment loss during its first transmission). No other segments are lost during transmissions.

What are the values for cwnd and ssthreshold when the sender receives the TCP ACK with number 15? Show your intermediate steps or your diagram in your solution.

Problem 5 continued on next page. . .

Write your solution to Problem 5 in this box



When ACK 15 is received, cwnd = 4 and ssthresh = 3.