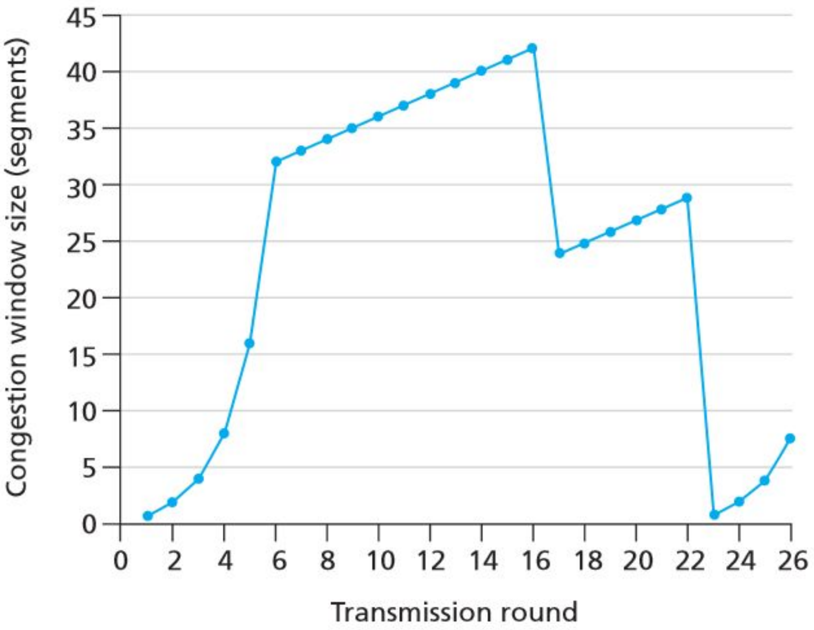
EECS 489 Discussion 5

Wireshark Demo

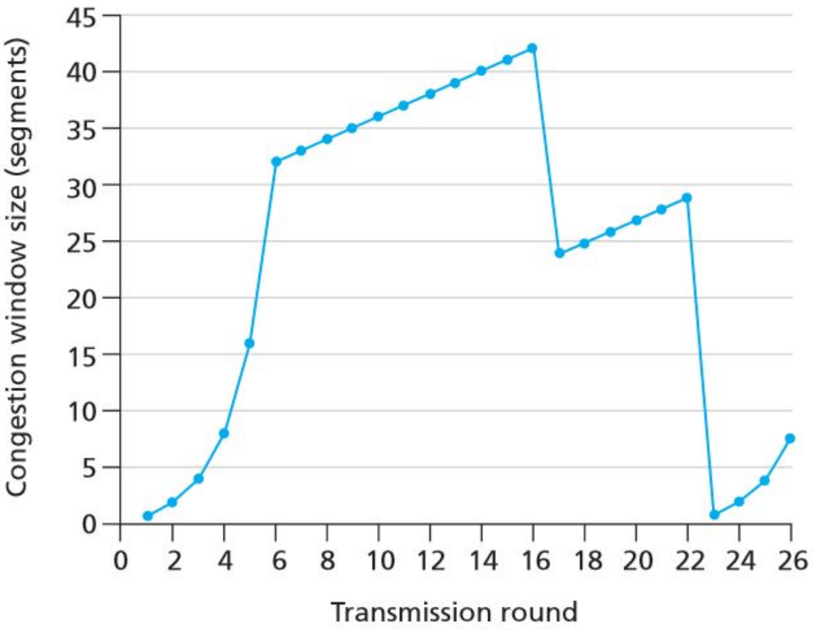
Q1



Assume TCP Reno is experiencing the behaviour shown to the left

- Identify the intervals of time when TCP slow start is operating
- Identify the intervals of time when TCP congestion avoidance is operating.
- After the 16th transmission round, is segment loss detected by a triple duplicate ACK or by a timeout?
- After the 22nd transmission round, is segment loss detected by a triple duplicate ACK or by a timeout?

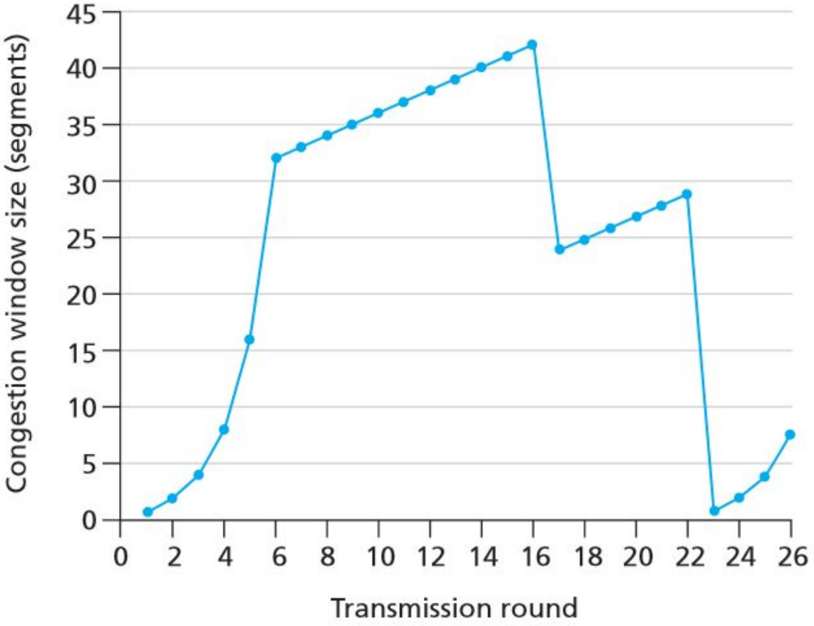
Q1



Assume TCP Reno is experiencing the behaviour shown to the left

- Identify the intervals of time when TCP slow start is operating [1,6] [23,26]
- Identify the intervals of time when TCP congestion avoidance is operating. [6,22]
- After the 16th transmission round, is segment loss detected by a triple duplicate ACK or by a timeout? Dup ACK
- After the 22nd transmission round, is segment loss detected by a triple duplicate ACK or by a timeout? Timeout

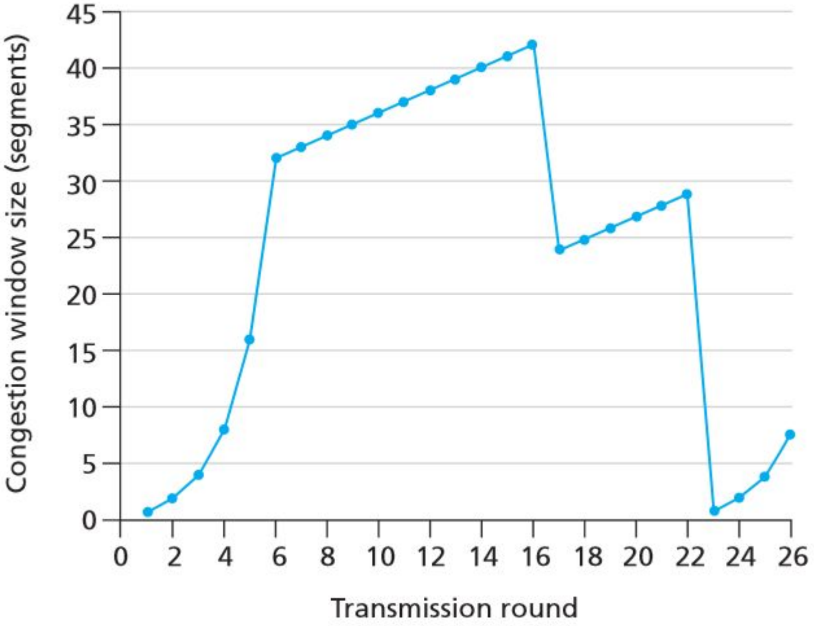
Q1



Assume TCP Reno is experiencing the behaviour shown to the left

- What is the initial value of ssthresh at the first transmission round?
- What is the value of ssthresh at the 18th transmission round?
- What is the value of ssthresh at the 24th transmission round?
- During what transmission round is the 70th segment sent?

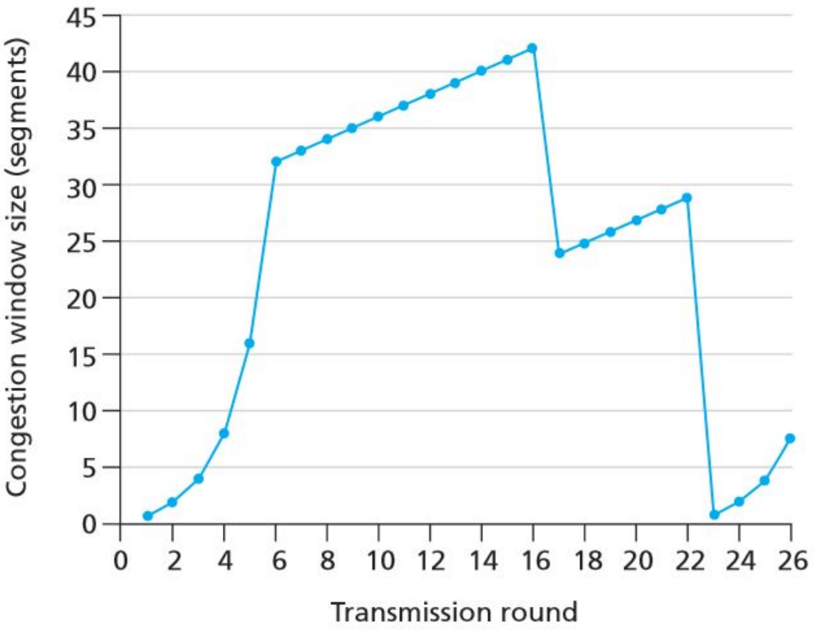
Q1



Assume TCP Reno is experiencing the behaviour shown to the left

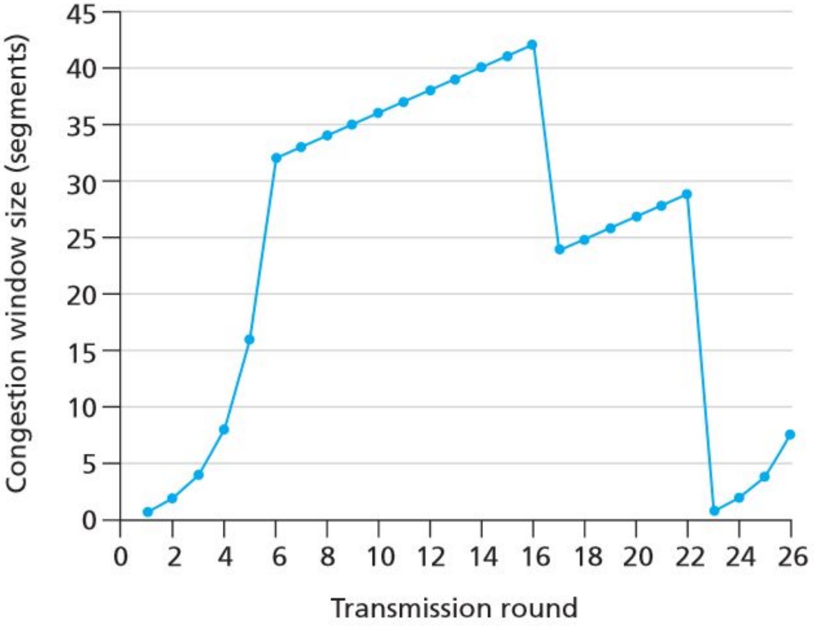
- What is the initial value of ssthresh at the first transmission round? 32
- What is the value of ssthresh at the 18th transmission round? 21
- What is the value of ssthresh at the 24th transmission round? 14
- During what transmission round is the 70th segment sent? 7

Q1



- Assuming a packet loss is detected after the 26th round by the receipt of a triple duplicate ACK, what are the values of the congestion window size and of ssthresh?
- Suppose TCP Tahoe is used, and assume that triple duplicate ACKs are received at the 16th round. What are the ssthresh and the congestion window size at the 19th round?
- Suppose TCP Tahoe is used, and there is a timeout event at 22nd round. How many packets have been sent out from 17th round till 22nd round, inclusive?

Q1



- Assuming a packet loss is detected after the 26th round by the receipt of a triple duplicate ACK, what are the values of the congestion window size and of ssthresh? 4,4
- Suppose TCP Tahoe is used, and assume that triple duplicate ACKs are received at the 16th round. What are the ssthresh and the congestion window size at the 19th round? 21, 4
- Suppose TCP Tahoe is used, and there is a timeout event at the start of the 22nd round. How many packets have been sent out from 17th round till 22nd round, inclusive? 1+2+4+8+16+21=52

Q2

Consider sending a large file from a host to another over a TCP connection that has no loss.

- Suppose TCP uses AIMD for its congestion control without slow start. Assuming cwnd increases by 1 MSS every time a batch of ACKs is received and assuming approximately constant round-trip times, how long does it take for cwnd increase from 6 MSS to 12 MSS (assuming no loss events)?
- What is the average throughput (in terms of MSS and RTT) for this connection up through time = 6 RTT (assume cwnd starts at 5 MSS)?

Q2

Consider sending a large file from a host to another over a TCP connection that has no loss.

- Suppose TCP uses AIMD for its congestion control without slow start. Assuming cwnd increases by 1 MSS every time a batch of ACKs is received and assuming approximately constant round-trip times, how long does it take for cwnd increase from 6 MSS to 12 MSS (assuming no loss events)? 6 RTT
- What is the average throughput (in terms of MSS and RTT) for this connection up through time = 6 RTT (assume cwnd starts at 5 MSS)? 7.5 MSS / RTT

Q3

Host A is sending an enormous file to Host B over a TCP connection. Over this connection there is never any packet loss and the timers never expire. Denote the transmission rate of the link connecting Host A to the Internet by R bps. Suppose that the process in Host A is capable of sending data into its TCP socket at a rate S bps, where S = 10*R. Further suppose that the TCP receive buffer is large enough to hold the entire file, and the send buffer can hold only one percent of the file. What would prevent the process in Host A from continuously passing data to its TCP socket at rate S bps? TCP flow control? TCP congestion control? Or something else?

Q3

Host A is sending an enormous file to Host B over a TCP connection. Over this connection there is never any packet loss and the timers never expire. Denote the transmission rate of the link connecting Host A to the Internet by R bps. Suppose that the process in Host A is capable of sending data into its TCP socket at a rate S bps, where $S = 10 \cdot R$. Further suppose that the TCP receive buffer is large enough to hold the entire file, and the send buffer can hold only one percent of the file. What would prevent the process in Host A from continuously passing data to its TCP socket at rate S bps? TCP flow control? TCP congestion control? Or something else? Bottleneck is connection to the internet. Send buffer is always full

Q4

Consider a single TCP Reno connection uses one 10 Mbps link which does not buffer any data. Suppose that this link is the only congested link between the sending and receiving hosts. Assume that the TCP sender has a huge file to send to the receiver, and the receiver's receive buffer is much larger than the congestion window. We also make the following assumptions: each TCP segment size is 1,500 bytes; the two-way propagation delay of this connection is 150 msec; and this TCP connection is always in congestion avoidance phase (ignore slow start).

- What is the maximum window size (in segments) that this TCPconnection can achieve?
- How long would it take for this TCP connection to reach its maximum window again after recovering from a packet loss (assume 3 dupACK's received)

Q4

Consider a single TCP Reno connection uses one 10 Mbps link which does not buffer any data. Suppose that this link is the only congested link between the sending and receiving hosts. Assume that the TCP sender has a huge file to send to the receiver, and the receiver's receive buffer is much larger than the congestion window. We also make the following assumptions: each TCP segment size is 1,500 bytes; the two-way propagation delay of this connection is 150 msec; and this TCP connection is always in congestion avoidance phase (ignore slow start).

- What is the maximum window size (in segments) that this TCPconnection can achieve? W*MSS/RTT = W*1500 bytes / 150 msec = 10 Mbps => 125
- How long would it take for this TCP connection to reach its maximum window again after recovering from a packet loss (assume 3 dupACK's received)? 9.375 sec