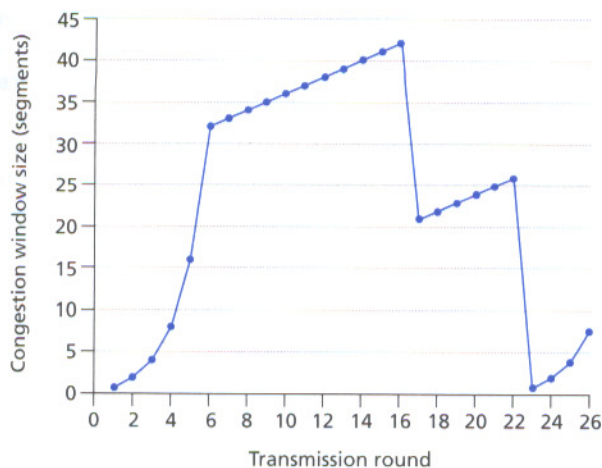


- a. What is the maximum value of  $L$  such that TCP sequence numbers are not exhausted? Recall that the TCP sequence number field has 4 bytes.
  - b. For the  $L$  you obtain in (a), find how long it takes to transmit the file. Assume that a total of 66 bytes of transport, network, and data-link header are added to each segment before the resulting packet is sent out over a 10 Mbps link. Ignore flow control and congestion control so A can pump out the segments back to back and continuously.
- P24. Host A and B are communicating over a TCP connection, and Host B has already received from A all bytes up through byte 248. Suppose Host A then sends two segments to Host B back-to-back. The first and second segments contain 40 and 60 bytes of data, respectively. In the first segment, the sequence number is 249, the source port number is 503, and the destination port number is 80. Host B sends an acknowledgement whenever it receives a segment from Host A.
- a. In the second segment sent from Host A to B, what are the sequence number, source port number, and destination port number?
  - b. If the first segment arrives before the second segment, in the acknowledgement of the first arriving segment, what is the acknowledgment number, the source port number, and the destination port number?
  - c. If the second segment arrives before the first segment, in the acknowledgement of the first arriving segment, what is the acknowledgment number?
  - d. Suppose the two segments sent by A arrive in order at B. The first acknowledgement is lost and the second acknowledgement arrives after the first timeout interval, as shown in the diagram on the next page. Draw a timing diagram, showing these segments and all other segments and acknowledgements sent. (Assume there is no additional packet loss.) For each segment in your figure, provide the sequence number and the number of bytes of data; for each acknowledgement that you add, provide the acknowledgment number.
- P25. Host A and B are directly connected with a 200 Mbps link. There is one TCP connection between the two hosts, and Host A is sending to Host B an enormous file over this connection. Host A can send application data into the link at 100 Mbps but Host B can read out of its TCP receive buffer at a maximum rate of 50 Mbps. Describe the effect of TCP flow control.
- P26. SYN cookies were discussed in Section 3.5.6.
- a. Why is it necessary for the server to use a special initial sequence number in the SYNACK?
  - b. Suppose an attacker knows that a target host uses SYN cookies. Can the attacker create half-open or fully open connections by simply sending an ACK packet to the target? Why or why not?

- P27. Consider the TCP procedure for estimating RTT. Suppose that  $\alpha = 0.1$ . Let  $\text{SampleRTT}_1$  be the most recent sample RTT, let  $\text{SampleRTT}_2$  be the next most recent sample RTT, and so on.
- For a given TCP connection, suppose four acknowledgments have been returned with corresponding sample RTTs  $\text{SampleRTT}_4$ ,  $\text{SampleRTT}_3$ ,  $\text{SampleRTT}_2$ , and  $\text{SampleRTT}_1$ . Express  $\text{EstimatedRTT}$  in terms of the four sample RTTs.
  - Generalize your formula for  $n$  sample RTTs.
  - For the formula in part (b) let  $n$  approach infinity. Comment on why this averaging procedure is called an exponential moving average.
- P28. In Section 3.5.3 we discussed TCP's estimation of RTT. Why do you think TCP avoids measuring the  $\text{SampleRTT}$  for retransmitted segments?
- P29. What is the relationship between the variable  $\text{SendBase}$  in Section 3.5.4 and the variable  $\text{LastByteRcvd}$  in Section 3.5.5?
- P30. What is the relationship between the variable  $\text{LastByteRcvd}$  in Section 3.5.5 and the variable  $y$  in Section 3.5.4?
- P31. In Section 3.5.4, we saw that TCP waits until it has received three duplicate ACKs before performing a fast retransmit. Why do you think the TCP designers chose not to perform a fast retransmit after the first duplicate ACK for a segment is received?
- P32. Consider Figure 3.46(b). If  $\lambda'_{\text{in}}$  increases beyond  $R/2$ , can  $\lambda_{\text{out}}$  increase beyond  $R/3$ ? Explain. Now consider Figure 3.46(c). If  $\lambda'_{\text{in}}$  increases beyond  $R/2$ , can  $\lambda_{\text{out}}$  increase beyond  $R/4$  under the assumption that a packet will be forwarded twice on average from the router to the receiver? Explain.
- P33. Consider the following plot of TCP window size as a function of time.





Assuming TCP Reno is the protocol experiencing the behavior shown above, answer the following questions. In all cases, you should provide a short discussion justifying your answer.

- a. Identify the intervals of time when TCP slow start is operating.
  - b. Identify the intervals of time when TCP congestion avoidance is operating.
  - c. After the 16th transmission round, is segment loss detected by a triple duplicate ACK or by a timeout?
  - d. After the 22nd transmission round, is segment loss detected by a triple duplicate ACK or by a timeout?
  - e. What is the initial value of **Threshold** at the first transmission round?
  - f. What is the value of **Threshold** at the 18th transmission round?
  - g. What is the value of **Threshold** at the 24th transmission round?
  - h. During what transmission round is the 70th segment sent?
  - i. Assuming a packet loss is detected after the 26th round by the receipt of a triple duplicate ACK, what will be the values of the congestion window size and of **Threshold**?
- P34. Refer to Figure 3.55, which illustrates the convergence of TCP's AIMD algorithm. 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 diagram similar to Figure 3.55.
- P35. In Section 3.5.4 we discussed the doubling of the timeout interval after a timeout event. This mechanism is a form of congestion control. Why does TCP need a window-based congestion-control mechanism (as studied in Section 3.7) in addition to this doubling-timeout-interval mechanism?
- P36. 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? Elaborate.
- P37. Consider sending a large file from a host to another over a TCP connection that has no loss.
- a. Suppose TCP uses AIMD for its congestion control without slow start. Assuming **CongWin** increases by 1 MSS every time a batch of ACKs is received and assuming approximately constant round-trip times, how long

does it take  
loss events

- b. What is the  
nection up

P38. Recall the ma  
from when th  
packet is lost

- a. Show that t

- b. Use the resu  
average rate

P39. In our discussi  
throughput of 1  
 $2 \cdot 10^{-10}$  (or equ  
Show the deriv  
and MSS value  
connection, wh

P40. In our discussi  
assumed that th  
that the TCP ser  
has no more dat  
of time and ther  
disadvantages o  
from  $t_1$  when sta  
mend? Why?

P41. In this problem  
of end-point aut  
a. Consider a se  
responds to th  
DNS server).  
Y, where will

- b. Now suppose  
the server into  
IP address?