Data Networks Summer 2007
Homework #3

Assigned June 18, 2007
Due June 25 in class

Name: ___________________________

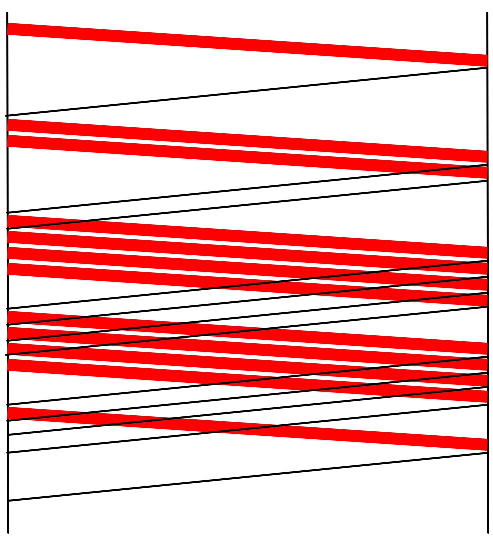
Email: ___________________________

Student ID: _______________________

Problem	Points
1	
2	
3	
4	
5	
Total	

Problem 1 (20 points) Host A is transferring a file of size L to host B using a TCP connection. Host A sends the file data in fixed size segments (or packets) equal to the Maximum Segment Size (MSS), a predetermined value, which gives the maximum number of data bytes that can be sent in a TCP segment. Host B sends an acknowledgement immediately upon receiving a data segment. Let R be the round trip delay between A and B. The transmission delay to send a data segment is T. Assume the transmission delay to send an acknowledgement is negligible. The advertised receiver window size of host B is W. In this problem, we are only concerned with the data transmission phase of TCP. We assume the TCP connection is already established. TCP performs the slow start and congestion avoidance mechanisms. There is no error or packet loss during transmission.

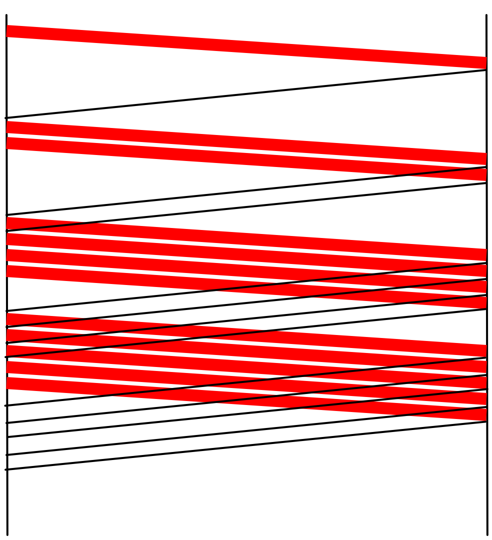
(a) Given W = 4*MSS, L = 12*MSS, how long does it take for the file to be sent and acknowledged? (10 points)



Advertised receiver window limits how large the sender's window can grow to. Therefore, when slow start opens up the sender's window size to W, the sender's window size will become fixed at W unless packet loss occurs. Here, we will also assume R to be larger than 5*T.

As can be seen, the total time required is 5*(T+R).

(b) Given W = 5*MSS, L = 12*MSS, how long does it take for the file to be sent and acknowledged? (10 points)



Now, since W is 5*MSS, the situation is shown in the left.

The total time is now 4*(T+R) + 4*T.

Problem 2 (20 points) TCP congestion control has many undesirable properties in practice. In the following scenarios, explain in detail what undesirable characteristics may a TCP data transfer exhibit and the underlying reasons for them.

(a) A single TCP connection on a link with 5 Gbps of bandwidth and the RTT is 100ms. (6 points)

At this bandwidth and RTT, it takes a window size of 62.5MB (5Gbps * 100ms) to fully utilize the link. TCP slow start would take many round trip times to reach a window size of 62.5MB because the beginning window size is one segment and a segment is usually small (approx. 1500 bytes). Thus, TCP's efficiency over a high bandwidth high delay link is very poor.

(b) A TCP connection with a 200ms RTT sharing a bottleneck link with 5 other TCP connections with a 5ms RTT. (7 points)

Since TCP's throughput, as we have analyzed in class, is inversely proportional to the RTT of the connection, TCP will provide unfair allocation of bandwidth among the 6 data flows in this scenario.

(c) A TCP connection over a wireless link where random bit errors may occur due to interference. (7 points)

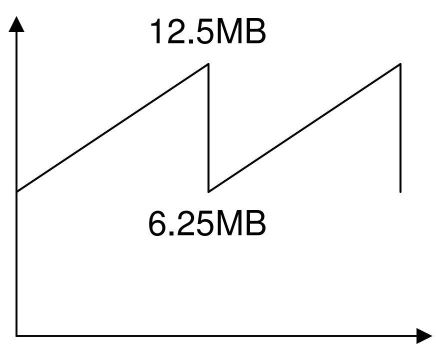
Since TCP uses packet loss as an implicit indicator of congestion and cut back the sender's window size to slow down, when a random bit error occurs in the wireless transmission and corrupt a packet, TCP would mis-interpret that as a sign of congestion and reduce sending window size by half, even though there is no congestion in the network.

Problem 3 (20 points) Consider a network link with the following characteristics: Link speed = 1Gbps, RTT = 100ms, and Data packet size = 1000 bytes. Assume all traffic is transmitted using a TCP-like window-based transmission protocol with the following congestion control algorithm: (1) Congestion avoidance, cwnd = cwnd + 1 after one RTT, and (2) Fast recovery, cwnd = cwnd/2 when loss occurs.

(a) What is the congestion window size W_0 such that when the traffic is transmitted using this window size, the network link is fully utilized? (6 points)

W_0 = Links speed * RTT = 12.5MB

(b) Suppose exactly when the congestion window size W reaches W_0, a loss occurs and the sender immediately (assume the loss is instantaneously known by the sender) activates the fast recovery algorithm. What is the asymptotic average throughput of this transmission? (7 points)



cwnd will oscillate between 12.5MB and 6.25MB. Since for each RTT, cwnd only grows by one packet (1KB), it takes 6250 RTTs to reach the peak cwnd size.

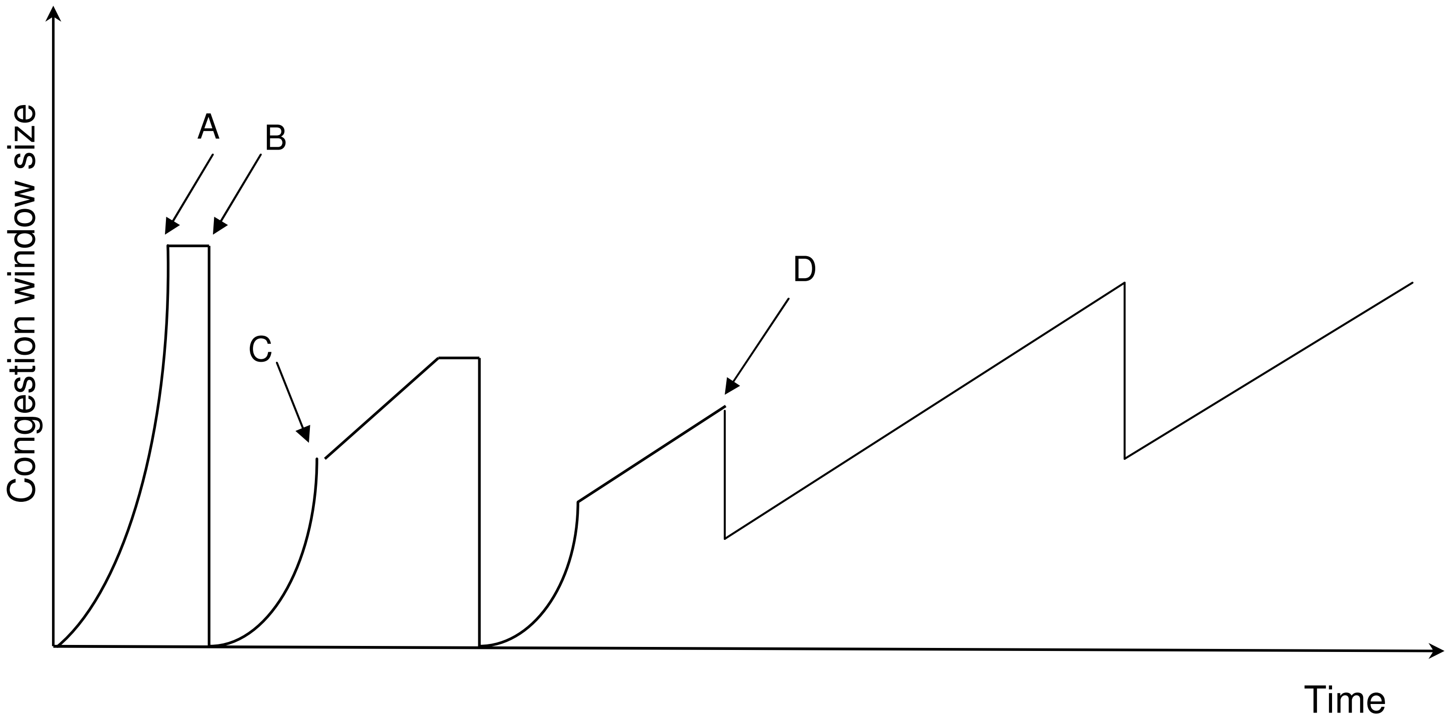
Average rate = total number of bytes sent per cycle / cycle time
=
(6250*6.25MB (the bottom rectangular area)
+ (1+2+…+6249)*1000) (the upper triangle area)
/ (6250*100ms) (cycle time)
= 749.96Mbps

(c) With the same assumptions made in (b), compute the time it takes for the flow to recover from a loss, that is the time it takes to increase its congestion window size to W_0 after a loss. (7 points)

That is simply 6250 RTTs, i.e. 625 seconds.

BTW, it is okay if you used the TCP model equation to calculate the answer for (b)

Problem 4 (20 points) Below is a graph showing the changes in the congestion windows size of a TCP Reno connection over time. There are several places in this graph where the congestion window size changes abruptly. In particular, 4 points have been labeled below using A to D. Explain what happened to the TCP Reno connection at each of the labeled points in the graph that caused the abrupt changes in congestion window size.



A: Packet loss occurred during slow start, no more new acknowledgement is received, so congestion window stops growing
B: Packet loss causes a timeout, congestion window is reset to 1, ssthresh is reset to half the previous congestion window size
C: Slow start opens the congestion window size to ssthresh, slow start stops, and congestion avoidance begins
D: Three duplicate acks were received, so fast recovery causes the the congestion window size to be halved, and the ssthresh is set to the same value. Congestion avoidance continues.

Problem 5 (20 points): Suppose two flows A and B are arriving at a weighted fair queuing scheduler. For simplicity, the link capacity is 10 bits per second. The two flows have equal weight. The arrival times, and packet sizes are shown in the tables below.

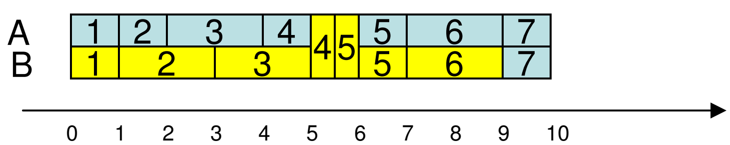
Flow A

Packet #	Arrival time (second)	Pkt size (bits)
1	0	5
2	1	5
3	1.5	10
4	1.5	5
5	6	5
6	6.5	10
7	8	5

Flow B

Packet #	Arrival time (second)	Pkt size (bits)
1	0	5
2	0.5	10
3	2	10
4	3	5
5	4	10
6	5	10
7	6	5

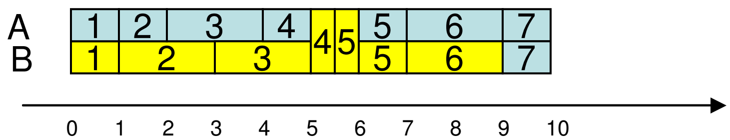
(a) Compute the start time and the finish time of every packet in the fluid flow system. You may draw a fluid flow system picture to help illustrate your answers. (8 points)



(b) Write down the packet transmission order in the real packet system. Use "A.1" to denote the first packet of flow A. (4 points)

A.1, B.1 (interchangeable)
A.2, B.2, A.3
B.3, A.4 (interchangeable)
B.4, B.5, A.5
A.6, B.6 (interchangeable)
A.7, B.7 (interchangeable)

(c) Recall that the system virtual time V(t) is the number of rounds of service the WFQ server has given at time t. V(t=0) = 0. One round of service is provided when 1 bit of service is given to every flow that has traffic to send. When a packet of flow A arrives at the system at time t, its <u>virtual start time</u> is either the current system virtual time V(t) or the virtual finish time of the previous packet in flow A, whichever is larger. For this problem, the <u>virtual finish time</u> of a packet is its virtual start time plus the packet's size in bits. Compute the virtual start time and the virtual finish time of every packet in the system. (8 points)



Flow A

Packet #	Virtual S	Virtual F
1	0	5
2	5	10
3	10	20
4	20	25
5	35 (V(t))	40
6	40	50
7	50	55

Flow B

Packet #	Virtual S	Virtual F
1	0	5
2	5	15
3	15	25
4	25	30
5	30	40
6	40	50
7	50	55