Homework 4.

Due Friday, March 25 at 11:59pm

EECS 325: 38 points + 3 bonus; EECS 425: 41 points

1. Consider host A connected to host B with a network path with bandwidth of 10Mbps and RTT of 100ms. Let A send a very large file to B over TCP, so that A always has data to send to B. Assume there is no other traffic in the network. Assume B's application consumes arriving data very quickly as soon as the data arrives, so that the TCP's receive buffer at host B is always empty. Further assume that host B does not use delayed ACKs (i.e., acknowledges every segment as soon as it arrives); that host A uses the initial value of congestion window threshold 64KB for new connections, and that it uses MSS is 1KB. Assume host A starts slow start from congestion window cwnd = 1MSS = 1KB. **(Hint: consider the format of TCP header and in particular the size of relevant field(s))**

   a. Draw the graph of how the value of cwnd at host A changes with time (assuming host A commences its transmission at time 0). Your graph must have time on X-axis and cwnd values on Y-axis. Make sure you indicate actual values on each axis (in other words, qualitative graph is not enough: please put coordinate values for key points of the graph). **(3 pts)**

   b. What is the value of cwnd at 1 hour after the beginning of transmission? **(3 pts)**

2. Assume a TCP sender has 25 segments ready to transmit to the receiver. How quickly the receiver is able to consume data is unknown. No delayed acknowledgements are employed, but the rest of TCP optimizations, including fast retransmit and fast recovery, are used. Consider the graph below, where each square indicates a packet sent by the TCP sender to the receiver (with X coordinate representing time of the packet departure and Y coordinate indicating the sequence number of the packet) and each triangle indicates an arrival of an acknowledgement at the sender (which X coordinate representing the time of arrival and Y coordinate indicating the sequence number of the acknowledged packet). (This type of graph is used quite often in analyzing dynamics of TCP: you record a trace of packets at a host and them plot a graph like this to understand what's going on.) Note, for simplicity we assume that sequence numbers count segments (rather than bytes), and acknowledgements carry the sequence number of the segment being acknowledged (rather than the sequence number of the next expected packet as in reality). Also assume that if congestion control needs to divide the sending rate by two, and the current flight size is an odd number, you just use integer division (i.e., 5:2 = 2). (Note: your answers to this problem include both textual explanations and scribbles on the figure. Please cut and paste and then print the figure on a separate sheet of paper; then scribble on it and scan it in as part of your answers to this problem.)

   (a) Show two approximate RTT intervals on the figure. **(2 pts)**

   (b) Indicate (on the picture) the time when the sender switches from slow start to congestion control. Explain your reasoning (if it's impossible to pinpoint the exact time, indicate the time range when the transition from slow start to congestion avoidance could plausibly occur) **(3 pts)**

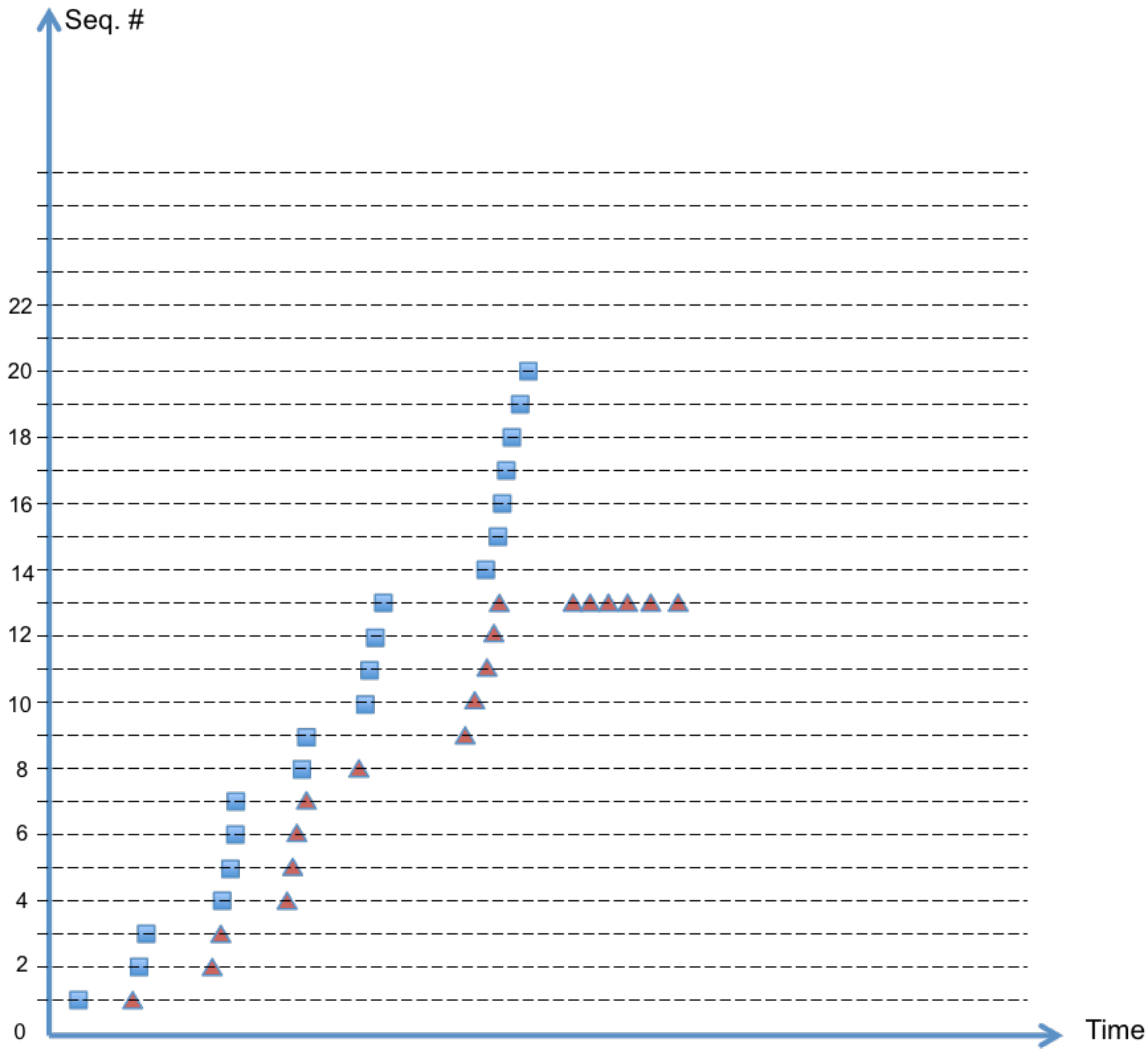
   (c) The graph shows that after the arrival of the ACKs 4-7, the sender only sent two segments (# 8 and 9) and then stalled until the arrival of the next ACK. Explain what could have caused this behavior. **(3 pts)**

   (d) After the arrival of a single ACK 8, the sender sent a burst of four packets. Give a scenario that could have caused this behavior. **(3 pts)**

   (e) One can tell from this graph that the sender will infer that there was at least one lost segments during the transmission. Which one (provide segment number)? Explain how could you tell. **(3 pts)**

   (f) The graph stops after transmission of segment 20. Complete the graph by showing the transmission of the remaining segments, until all 25 segments are reliably transferred to the receiver. Make sure you indicate the retransmission of the lost segment identified in sub-problem (e). Assume that that was the only lost segment. **(3 pts)**

   (g) Superimpose on this figure the graph reflecting how the values of congestion windows change with time (pick the time of transition from slow star to congestion avoidance from the range you found in sub-problem (b)). **(3 pts)**



3. TCP protocol standardizes header field sizes so that it would be suitable for a large range of different networks. But assume you are to design a TCP-style protocol for a specific pair of hosts, connected via a 100Mbps network path with RTT of 100ms. The protocol would behave exactly like TCP except you have freedom to change the header field sizes. How many bits should you allocate for the advertised window and sequence number fields in the segment header of your protocol assuming the maximum segment lifetime (i.e., the time you need to allow a segment to wonder around before showing up at the receiver) is 60 seconds? **(3 points)**

4. Let us assume some alternative flavors of congestion control.

    a. Assume a congestion control algorithm that still uses additive increase and multiplicative decrease, but instead of reducing sending rate by have after a detected packet loss, the sender would decrease it by ¼. Will this new control converge to fairness under the same assumptions as in the TCP fairness analysis in slide 3 of Lecture notes 13? Why or why not? (Use diagrams similar in that slide, also in Fig. 3.56 in the book, to argue.) **(2 points)**

    b. Assume a new congestion control algorithm that uses additive-increase and also additive decrease, where the sending rate is decreased after a loss by a constant amount, equal to the amount by which it is increased when there is no loss. Will this new control converge to fairness under the same assumptions as in the TCP fairness analysis in slide 3 of Lecture notes 13? Why or why not? (Use diagrams similar in that slide, also in Fig. 3.56 in the book, to argue.) **(2 points)**

5. We saw that assuming the network does not delay segments for too long (basically, as long as the maximum segment life is shorter than the sequence number wrap-around time), GBN needs at least N+1 distinct sequence numbers while SR needs 2N, where N is the window size. TCP inherits ideas from both GBN and SR. How many distinct sequence numbers TCP needs under the same assumptions (i.e., that as the maximum segment life is shorter than whatever the sequence number wrap-around time we decide to have, so the maximum segment life is not an issue), and also assuming the window that maximizes the sequence number space? **(3 points)**

EECS 425 only (Bonus for 325):

6. Explain as precisely as you can how delayed acknowledgements will affect the way the congestion window changes during the congestion avoidance phase, assuming the congestion control algorithm as discussed in class. **(3 pts)**