

CS 484/504 HOMEWORK - IV

Fall 2022

Graduate Level Course Code CS 504
Computer Science Department, New Mexico State University

For graduate students ($10 \times 10 = 100$)

For undergraduate students ($8 \times 12.5 = 100$)

For undergraduate students ($2 \times 12.5 = 25$) for extra credit questions 4 and 8

1. What do we mean when we say that TCP is self-clocking? Why is TCP self clocking?
2. Why do TCP and UDP messages use ports? Describe a situation where two computers might need to have two separate TCP conversations running at the same time.

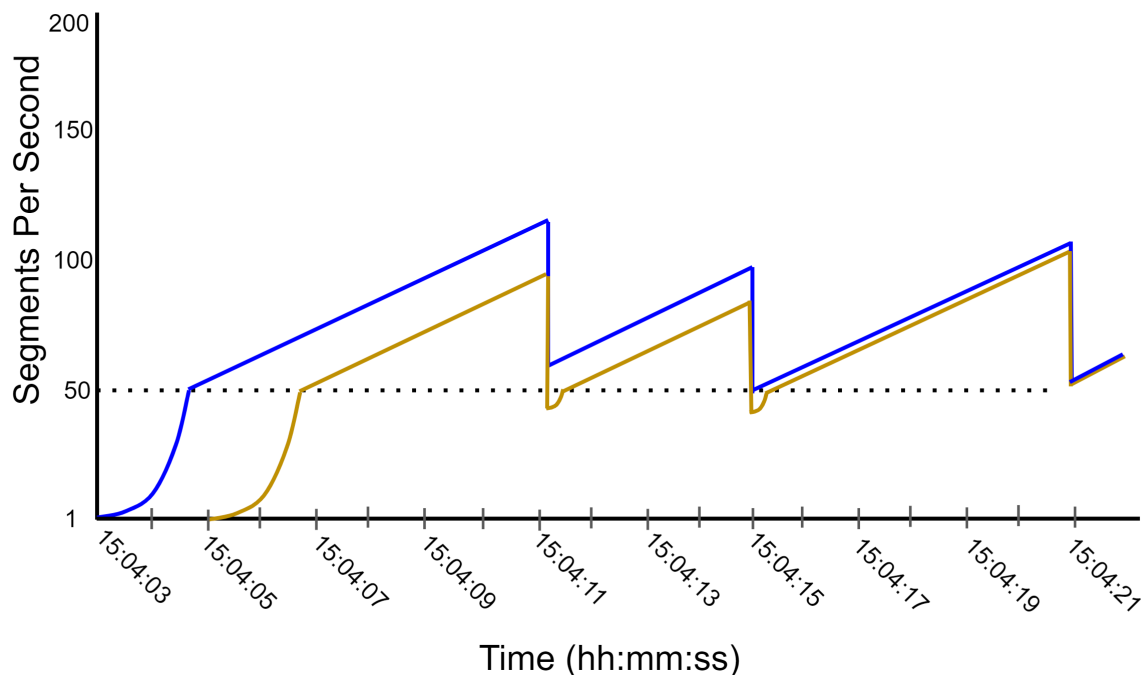


Figure 1: **Transmission speed over time of two TCP conversations. The first (orange) begins at 15:04:03 and the second (blue) begins at 15:14:05.**

3. Consider the diagram in Figure 1. Answer with at least a full sentence.
 - (a) Which TCP flavor is being used for each of these TCP connections? How can you tell?
 - (b) At what times do these TCP conversations detect congestion?
 - (c) How did these two TCP conversations know there was congestion? Reference the self-clocking property of TCP in your answer.
4. Consider the diagram in Figure 1. Answer with at least a full sentence.
 - (a) The transmission speeds of these two conversations are getting closer and closer together as time goes on. Why is that happening?
 - (b) If a third TCP conversation started up using the exact same TCP flavor with the same rules, would the same thing happen? Why or why not?

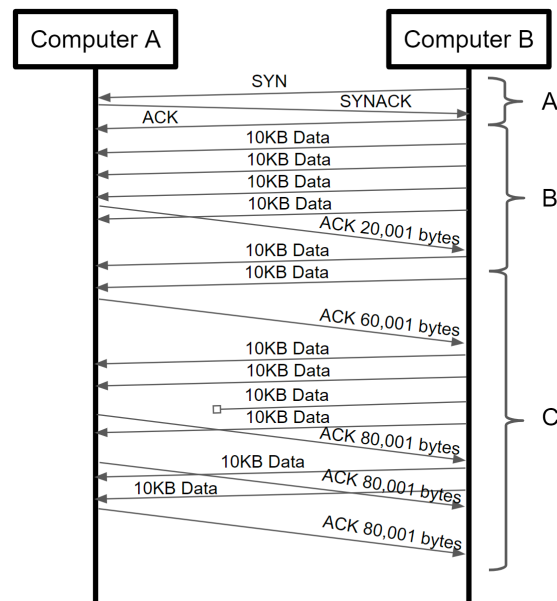


Figure 2: **The beginning of a TCP conversation.**

5. Consider the diagram in Figure 2.
 - (a) What is happening in the section labeled “A”?
 - (b) From looking at the section labeled “B”, what would you guess that Computer B’s sending window size is in Kb? Why?
 - (c) From looking at the section labeled “B”, what would you guess that Computer A’s receiving window size is in Kb? Why?
6. Consider the diagram in Figure 2.
 - (a) What goes wrong in the section labeled “C”?
 - (b) How does Computer B know something has gone wrong?
 - (c) What should Computer B do next to fix the problem?
7. Imagine you are working for NMSU ICT’s networking team and you notice that suddenly there are more TCP retransmissions than transmissions flowing through the campus network. What is happening and why is this bad?
8. Draw a TCP packet diagram like Figure 2 where the following things happen:
 - A three-way handshake starts the TCP session
 - Computer A sends data and it is acknowledged
 - Computer B sends data and it is acknowledged
 - A polite TCP session termination
9. **Graduate Question:** Create a diagram similar to Figure 1 where:
 - Two TCP conversations start within two seconds of each other and experience congestion every 2-6 seconds.

- Your diagram must follow these two conversations for at least 30 seconds.
- The first machine must use TCP Reno with a Slow Start threshold of 50 segments per second. The additive increase is 10 segments per second per second. The multiplicative decrease is a division by 2.
- The second machine must also use TCP Reno with a Slow Start threshold of 50 segments per second. The additive increase is 20 segments per second per second. The multiplicative decrease is a division by 4.

Use a straightedge and/or add grid lines as necessary to make sure you are accurate in your drawing. You will need an accurate picture for the next question.

10. **Graduate Question:** Return to your diagram from the previous question. There is more to the difference between TCP implementations than just its flavor.
- (a) Will these two TCP conversations eventually converge on a fair sharing of the available bandwidth? Why or why not?
 - (b) You can determine how many segments of data were sent in your diagram by integrating (as in calculus area under-the-curve) each of these lines. How many segments of data did each of your TCP conversations communicate? A rough geometric approximation is okay here.
 - (c) Which conversation transmitted more segments of data? Why?
 - (d) Now do you believe that the tools you learned in calculus are actually useful for computer scientists? :)