Lab Assignment 1

CSC317: Computer Networks

Submission instructions: Submit the Python source file containing your code in Moodle. You MUST show me your work in person to receive credit for this assignment.

In this lab, we will discuss how a discrete event simulator may be used to compute the queueing times when the math becomes complicated. The simulator that we present in the lab is available in Moodle.

In this assignment, you will be extending the simulator for the topology shown in Figure 1. Nodes A and B generate packets destined for node E that must pass through nodes C and D. The challenge in simulating this topology is that each node has a single queue that must be shared by all the packets processed (i.e. sent or received) by the node. To illustrate the behavior of the shared queue, consider the scenario when a packet from node A has already arrived at node C. While node C is transmitting the packet from node A, a packet from node B arrives. The packet from node B cannot be transmitted until node C completes transmitting the packet from node A. This type of behavior is common for both routers and switches.

We assume that the queues used by nodes C and D are finite and any packet approaching them is dropped when the corresponding queue is full. Upon receiving a packet, nodes C and D send an acknowledgement (ACK) to the sender if the packet is not dropped, and send a negative-acknowledgement (NAK) otherwise. Node E never drops any packet and always sends ACK when a packet is received. Any node transmitting a packet always transmits the packet at the head of the queue (HOQ), and waits for an ACK/NAK message from the receiver. If an ACK is received, the sender pops the packet at the HOQ and transmits the next packet (i.e. the new HOQ) from queue. The node retransmits the packet at the HOQ if a NAK is received. Your task is to modify the provided simulator to account for this behavior.

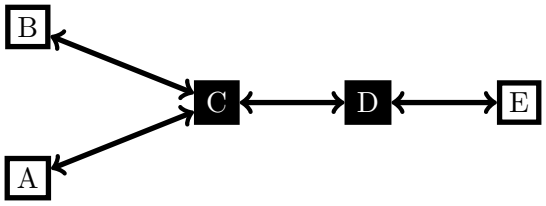

Figure 1: Network topology

For this simulation, we assume that an application on node A enqueues (for transmission) a bursts of 20 packets in every 1000 simulation ticks. Similarly, an application running on Node B enqueues a bursts of 5 packets in every 500 simulation ticks. The transmission and the propagation delays for the link between D and E are 20 ticks and 5 ticks, respectively. The transmission and the propagation delays for all other links are 10 ticks and 1 tick, respectively. Same transmission and propagation delays apply for a ACK/NAK message. Each of the finite queues in nodes C and D can hold at most 35 packets at a time, and the queues in nodes A and B are infinite. The simulation runs for 10,000 ticks.

The output of the simulation will be a CVS file named "sim.csv" containing the following table:

SeqN.	Src	Q@src	Trn@src	Rcv@C	Drp@C	Trn@C	Rcv@D	Drp@D	Trn@D	Rcv@E

The table includes the following information for each of the packets that were generated during the simulation.

1. SeqN: a unique sequence number of the packet
2. Src: the source of the packet (i.e., either node A or B)
3. Q@src: time when a packet was queued at the source (i.e., either node A or B)
4. Trn@src: time when a packet was transmitted by the source
5. Rcv@C: time when a packet was received at node C
6. Drp@C: is the packet dropped by node C (Yes/No)
7. Trn@C: time when a packet was transmitted by node C
8. Rcv@D: time when a packet was received by node D
9. Drp@D: is the packet dropped by node D (Yes/No)
10. Trn@D: time when a packet was transmitted by node D
11. Rcv@E: time when a packet was received by node E

In case of a packet drop, the remaining fields of the table should contain `None`.

Your code should run in **Python 3**. No additional packets beyond the standard distribution should be used. Additionally, based on the data generated by the simulator, answer the following questions:

1. What was the average end-to-end delay of each packet?
2. What was the average queueing delay of each packet?
3. If you are asked to replace only one link in the network, which link will you change to minimize the average end-to-end delay? Why?