Assignment 3

Student’s name:
1 Preview

This assignment builds on your experiences with Assignments 1 and 2. First, you will enhance the client and server to incorporate fair delay-driven congestion control. Second, you will change the FIFO forwarder from Assignment 2 to implement SRPT (Shortest Remaining Processing Time) scheduling of messages for transmission into the emulated bottleneck link. Then, you will experiment with four sessions of the congestion control protocol: Session 1 starts at time 0 and transmits 5,000 messages; Session 2 starts at time 10 seconds and transmits 3,000 messages; Session 3 starts at time 30 seconds and transmits 2,000 messages; Session 4 starts at time 75 seconds and transmits 2,000 messages. You will conduct the experiment first with the original FIFO forwarder and then with the alternative SRPT forwarder. For both forwarders, the emulated bottleneck link capacity is 1 Mbps.

2 Client and Server Implementation

Change your implementation of the application-level over-UDP client and server programs as follows. The client transmits messages of size 1,000 bytes to the server through the forwarder. The client stamps each message with time of its transmission. The client controls transmission using a window sized to \( W \) messages, i.e., transmits up to \( W \) unacknowledged messages. Initially, the window allows one unacknowledged message, i.e., \( W \leftarrow 1 \). Upon receiving a message, the server responds immediately with a small ACK that echoes the timestamp from the received message.

To update the window, the client maintains counters \( C \) and \( X \). Upon initialization, both counters are set to 0, i.e., \( C \leftarrow 0 \) and \( X \leftarrow 0 \). Upon receiving an ACK, the client calculates round-trip time \( T \) of the message-ACK pair as the difference between current time and the timestamp returned by the ACK. If \( T \) is calculated for the first time during the session, or \( T \) is less than the minimum round-trip time \( M \) that has been observed so far (i.e., if \( T < M \)), the calculated \( T \) becomes the value of \( M \), i.e., \( M \leftarrow T \). Then, the client performs one of the following:

- If \( T > M + 80 \text{ ms} \) and \( X \leq 0 \), the client sets the \( X \) counter to \( W - 1 \), resets the \( C \) counter, and reduces the window size to the maximum of one and \( \lfloor \frac{3W}{4} \rfloor \) messages, i.e., \( X \leftarrow W - 1 \), \( C \leftarrow 0 \), and \( W \leftarrow \max\{\lfloor \frac{3W}{4} \rfloor, 1\} \).
- If \( T \leq M + 80 \text{ ms} \) and \( X \leq 0 \), the client increases the \( C \) counter by one (i.e., \( C \leftarrow C + 1 \)) and compares the new value with \( W \); if \( C \geq W \), the client increases the window size by one message and resets the \( C \) counter, i.e., \( W \leftarrow W + 1 \) and \( C \leftarrow 0 \).
- If \( X > 0 \), the client decreases the \( X \) counter by one but does not change the window size, i.e., \( X \leftarrow X - 1 \).

As in Assignment 1, the client enhances each message with two control information fields: 1) identifier \( S \) of the session and 2) number \( R \) of messages that the session still needs to transmit.

3 Alternative Forwarder Implementation

Referring to the forwarder program from Assignment 2 as Forwarder A, implement the following alternative Forwarder B. Forwarder B maintains a separate FIFO queue for each session. When a message arrives, the forwarder checks field \( S \) in the message to identify to which session the message belongs; then the forwarder appends the message to the tail of the session queue. As with Forwarder A, Forwarder B waits for time \( t \) before sending the next pending message where \( t \) refers to the transmission delay of message \( m \) on the emulated 1-Mbps bottleneck link. If the waiting messages belong to multiple sessions, Forwarder B serves the queue with the smallest \( R \) in the head-of-the-queue message. The forwarder records changes of the message queue lengths for all four queues.

4 Experimental Study

The traffic is generated by the four sessions specified in Section 1. Conduct the experiment in both configurations, with Forwarder A or Forwarder B. Set the buffer sizes in the forwarders sufficiently high to avoid message losses. If losses do occur due to your congestion control implementation, fix the implementation to avoid losses. The forward (data) path of each session traverses the forwarder, which emulates the 1-Mbps bottleneck link. The reverse (acknowledgment) path of each session contains the delay plugin with the delay value set to 400 ms during the experiments. Place the server of each session in a separate host and monitor the bitrate of traffic coming to each of
the four servers. Hence, as in Assignment 2, ONL is used only to provide the desired network paths, delay the ACKs on the paths from the servers to the clients, and monitor the bitrates of the traffic arriving to the servers.

5 Report Preparation

Prepare a report that includes the following items:

1. Source code for the client, server, and both forwarder programs as well as instructions how to install and run the programs;

2. For Forwarder A:
   - Automatically plotted graph of the message queue length as a function of time in accordance with the measurements recorded by the forwarder;
   - Screen shot of the bitrates received by the servers of the four sessions, four lines on a single graph;

3. For Forwarder B:
   - Automatically plotted, single graph of the message queue lengths (four lines) as functions of time in accordance with the measurements recorded by the forwarder;
   - Screen shot of the bitrates received by the servers of the four sessions, four lines on a single graph;

4. Your analysis of the experimental results and their comparison with your theoretical expectations;

5. Your computation of session delays where delay $d_i$ of session $i$ is the amount of time passed since the client of the session transmits its first message until the client receives an ACK for its last message; report all four individual session delays $d_i$ and average session delay $D_i = \frac{1}{4} \sum_{i=1}^{4} d_i$; compare the average session delays for Forwarder A and Forwarder B.

The report must be submitted by email to gorinsky@arl.wustl.edu in PDF format, except for Item 1 which should be submitted as separate text attachments to the email message. The report is due on Wednesday, April 1, by 2:00 pm, i.e., a half an hour before the class. A hard copy of the report must be submitted to the instructor before the class starts.