Jason Lai
204995126
Dis 1A
CS 118
May 7, 2020

Homework 5

1 Problem 1

Host A and B are directly connected with a 100 Mbps link. There is one TCP connection between the two hosts, and Host A is sending to Host B an enormous file over this connection. Host A can send its application data into its TCP socket at a rate as high as 120 Mbps but Host B can read out of its TCP receive buffer at a maximum rate of 50 Mbps. Describe the effect of TCP flow control.

In accordance with TCP flow control, Host B allocates a receive buffer with a size denoted by a variable RcvBuffer that is set via socket options; this receive buffer is where data sent by Host A over the connection is received by Host B. Host B then sets its receive window size rwnd to amount of spare room in the receive buffer. However, since Host B can only read out of its receive buffer at a maximum rate of 50 Mbps and Host A can send data at a rate of 100 Mbps (up to 120 Mbps, but limited by 100 Mbps link), we see that the receive buffer will fill up faster than Host B can read out of it. The effect of TCP flow control in this case is that Host B's buffer will fill up as Host A sends data to Host B, so Host B must tell Host A when its buffer is full: it does this by placing the current value of the receive window size rwnd in the every segment it sends to Host A. When Host B's receive buffer is full, there is no spare room in the buffer, therefore the rwnd value would be 0 and subsequent segments sent to Host A would indicate that Host B has no more room in its receive buffer, so Host A should stop sending data. In order to know when it can start sending data again, Host A will continue sending arbitrary one byte segments to Host B, so that Host B can send ACK segments back to Host A to keep Host A updated on space in the buffer through the rwnd value. When the application at Host B reads data from the receive buffer and makes room in the buffer, the rwnd value in subsequent segments sent to Host A will then indicate that Host A can begin sending data again until Host B's buffer is full again or all data for the file has been sent. This process will repeat until all data for the entire file from Host A has been sent to Host B successfully.

2 Problem 2

A sends a TCP FIN message to B to close the TCP connection with B, the TCP header of A's FIN message is shown below. When B receives A's TCP FIN, it also decides to close the connection, so B sends a combined FIN and FIN-ACK message, whose TCP header is also shown below. Please fill in all the fields with a question mark in this TCP header.

Note*: Second (bottom) TCP header is the one that had question marks; filled-in answers are **bolded**.

<table>
<tr><th colspan="9">... src 1.1.1.1, dst: 2.2.2.2</th></tr>
<tr><td colspan="8">s_port: 2008</td><td>d_port: 5670</td></tr>
<tr><td colspan="9">seq_no: 980</td></tr>
<tr><td colspan="9">ack_no: 3120</td></tr>
<tr><td>header length</td><td>not used</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>rcv window: 200</td></tr>
<tr><td colspan="8">checksum: ...</td><td>0 (ignore this field)</td></tr>
</table>

<table>
<tr><th colspan="9">... src 2.2.2.2, dst: 1.1.1.1</th></tr>
<tr><td colspan="8">s_port: 5670</td><td>d_port: 2008</td></tr>
<tr><td colspan="9">seq_no: 3120</td></tr>
<tr><td colspan="9">ack_no: 981</td></tr>
<tr><td>header length</td><td>not used</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>rcv window: 400</td></tr>
<tr><td colspan="8">checksum: ...</td><td>0 (ignore this field)</td></tr>
</table>

Jason Lai
204995126
Dis 1A

3 Problem 3

Recall the macroscopic description of TCP throughput (Slide 134), in the period of time from when the connection's rate varies from W/(2 RTT) to W/RTT, only one packet is lost (at the very end of the period).

(a) Show that the loss rate (fraction of packets lost) is equal to L=lossrate= $1/(3/8W^2 + 3/4W)$

(b) Use the result above to show that if a connection has loss rate L, then its average rate is approximately given by $\simeq 1.22 \times MSS/(RTT \times \sqrt{L})$. (Hint: average rate $= \frac{3}{4} \cdot \frac{W}{RTT}$)

a) We have that the transmission rate increases by MSS for every RTT, until we have that $\frac{MSS}{RTT}$ is equal to $\frac{W}{2RTT}$, so that we have the initial lower bound transmission rate: $\frac{W}{2RTT}$ plus $\frac{W}{2RTT}$ gives the upper bound transmission rate: $\frac{W}{RTT}$. We disregard the RTT, since we are concerned with the number of packets sent within the period of time to find the fraction of packets lost. We can calculate the number of packets sent in the period of time from when the connection's rate varies from $\frac{W}{2RTT}$ to $\frac{W}{RTT}$ as follows:

$$\begin{aligned}
\sum_{i=0}^{\frac{W}{2}} \left(\frac{W}{2} + i\right) &= \sum_{i=0}^{\frac{W}{2}} \frac{W}{2} + \sum_{i=0}^{\frac{W}{2}} i \\
&= \frac{W}{2}\left(\frac{W}{2} + 1\right) + \frac{W}{2}\left(\frac{\frac{W}{2}+1}{2}\right) \text{ (note initial } i = 0) \\
&= \frac{W^2}{4} + \frac{W}{2} + \frac{W^2}{8} + \frac{W}{4} \\
&= \frac{2W^2}{8} + \frac{2W}{4} + \frac{W^2}{8} + \frac{W}{4} \\
&= \frac{3W^2}{8} + \frac{3W}{4} \text{ (this is the \# of packets sent within the period)}
\end{aligned}$$

From this, we have that if there is only one packet lost within this period of time, the loss rate is one divided by the number of packets sent within this period of time:

$$L = \frac{\text{\# of packets lost}}{\text{\# of packets sent}} = \frac{1}{\frac{3W^2}{8} + \frac{3W}{4}}$$

$\square$

b) For large W, we have that in the equation for number of packets in the period of time when the connection's rate varies $\frac{3W^2}{8} + \frac{3W}{4}$, the term $\frac{3W^2}{8} \gg \frac{3W}{4}$, therefore

$$L \approx \frac{1}{\frac{3W^2}{8}} \approx \frac{8}{3W^2} \text{ for when } W \text{ is large}$$

Solving for W, we have

$$W \approx \sqrt{\frac{8}{3L}}$$

To get the average rate, we have

$$\text{Average Rate} = \frac{3}{4} \cdot \frac{W}{RTT} \cdot MSS$$

where we multiply the average rate from the hint by MSS to get the average rate in terms of bytes of the maximum segment size.

To get the average rate when the connection has a loss rate L, we can substitute the value for W which we approximated for when W is large, so we have:

$$\begin{aligned}
\text{Average Rate} &= \frac{3}{4} \cdot \frac{W}{RTT} \cdot MSS \\
&= \frac{3}{4} \times \sqrt{\frac{8}{3L}} \times \frac{MSS}{RTT} \\
&= \frac{3}{4} \times \sqrt{\frac{8}{3}} \times \frac{MSS}{RTT \times \sqrt{L}} \\
&= 1.224744871 \times \frac{MSS}{RTT \times \sqrt{L}} \\
&\simeq 1.22 \times \frac{MSS}{RTT \times \sqrt{L}} \\
&\square
\end{aligned}$$

4 Problem 4

You are designing a reliable, sliding window, byte-stream protocol similar to TCP. It will be used for communication with a geosynchronous satellite network, for which the bandwidth is 800 Mbps and the RTT is 400 ms. Assume the maximum segment lifetime is 25 seconds.

(a) How many bits wide should you make the ReceiveWindow and SequenceNum fields? (ReceiveWindow is also called "Advertised Window" in some other textbooks.)

(b) If ReceiveWindow is 16 bits, what upper bound would that impose on the effective bandwidth?

a) The ReceiveWindow field should be 26 bits wide and the SequenceNum field should be 32 bits wide.

The receive window itself must be large enough to accommodate all data sent during a round trip time for a fully utilized connection; that is, if the the connection is fully utilized, the sender is constantly sending data, so the receive window must be large enough to handle data that is sent while other packets and ACKs are mid-flight to tell the sender to stop or continue sending data. This means we have to find the amount of data is sent between the first packet arriving and the first ACK arriving, which is the round trip time multiplied by the connection's bandwidth:

$$\text{Bandwidth} \times \text{RTT} = 800 \times 400 \times \frac{1\text{s}}{1000\text{ms}} = 320\,\text{Mb} = 40000000\,\text{Bytes}$$

The amount of data sent on a fully-utilized connection for this configuration before the first ACK arrives is 40000000 bytes, so we need a receive window that can hold at least that much, so, in bits, the ReceiveWindow field should be at least $\lceil \log_2 40000000 \rceil = 26$ bits.

We need enough sequence numbers so that the sequence wraps around when all possible packets from the old receive window are definitely gone, which is given by the maximum segment lifetime. So we need the total number of bytes of the segments that could exist at any given moment, which is given by the maximum segment lifetime multiplied by the connection's bandwidth:

$$\text{Bandwidth} \times \text{Lifetime} = 800 \times 25 = 20000\,\text{Mb} = 2500000000\,\text{Bytes}$$

The number of bits for the SequenceNum field must then be at least $\lceil \log_2 2500000000 \rceil = 32$ bits.

b) If the ReceiveWindow is 16 bits, this would impose an upper bound of ≈ 1.31072 Mbps on the effective bandwidth.

This is because, if the ReceiveWindow is 16 bits, then this means the receiver will only accommodate up to $2^{16} = 65536$ Bytes. This means that we need to limit the bandwidth so that the sender does not overflow the buffer at the receiver within a round trip time. This is limited bandwidth is given by:

$$\text{Bandwidth} = \frac{65536\text{Bytes}}{400\text{ms}} \times \frac{1000\text{ms}}{1\text{s}} \times \frac{0.000008\text{Mb}}{1\text{Byte}} \approx 1.31072 \text{ Mbps.}$$

Jason Lai
204995126
Dis 1A

5 Problem 5

Consider the evolution of a TCP connection with the following characteristics. Assume that all the following algorithms are implemented in TCP congestion control: slow start, congestions avoidance, fast retransmit and fast recovery, and retransmission upon timeout. If ssthresh equals to cwnd, use the slow start algorithm in your calculation.

- The TCP receiver acknowledges every segment in cumulative way, and the sender always has data segments available for transmission.
- The RTT is 100 ms for all transmissions, consists of the network latency of 60 ms in sending a segment (header and payload) from the sender to the receiver and 40 ms in sending an acknowledgment (header only) from the receiver to the sender. Ignore packet-processing delays at the sender and the receiver.
- Initially ssthresh at the sender is set to 5. Assume cwnd and ssthresh are measured in segments, and the transmission time for each segment is negligible.
- Retransmission timeout (RTO) is initially set to 500ms at the sender and is unchanged during the connection lifetime.
- The connection starts to transmit data at time t = 0, and the initial sequence number starts from 1. TCP segment with sequence number 6 is lost once (i.e., it sees segment loss during its first transmission). No other segments are lost during transmissions.

What are the values for cwnd and ssthreshold when the sender receives the TCP ACK with number 15? Show your intermediate steps or your diagram in your solution.

The values for cwnd and ssthreshold when the sender receives the TCP ACK with number 15 are:

$$\text{cwnd} = 4$$
$$\text{ssthreshold} = 3$$

Diagram on next page.

Homework 5

Jason Lai
204995126
Dis 1A

Cwnd	SSthresh	Algorithm
1	5	Slow start
2	5	Slow start
4	5	Slow start
6	5	Congestion Avoidance
6	5	Congestion Avoidance
6	5	Congestion Avoidance
$3+3=6$	$\max(2, \frac{6}{2}) = 3$	Fast Retransmit
$6+1=7$	3	Fast Recovery
cwnd = ssthresh = 3 ss after FR : $3+1=4$	3	Fast Recovery over
$4+\frac{1}{4} \approx 4$	3	Congestion Avoidance
$4+\frac{2}{4} \approx 4$ (15) $4+\frac{3}{4} \approx 4$ (16)	3	Congestion Avoidance