Computer Networking

Assignment 2

**Name: 陈闻天**

**Student number: 2023280259**

Problem 1: UPD (20 points)

（1）01010011+01100110=10111001，10111001+ 01110100=00101101，due to the carry, the sum of three 8-bits bytes should be 00101101+1=00101110，the 1s complement of 00101110 is 11010001.

（2）Receiver calculate the sum of all bytes including checksum and can detect errors if any bit of the sum is 0. Using 1s complement, the receiver just needs to add all bytes and finally check whether any bits of the sum is 0. If using the sum as checksum, the receiver needs to calculate all bytes excluding checksum, and finally check whether the sum just got is equal to the checksum. The add operation is simpler than the compare operation, so using the 1s complement of the sum as checksum is more efficient than using the sum as checksum.

(3) If the addition of the sum and check sum is not 0, the receiver will detect errors. It is impossible to miss 1-bit error, but it is possible for 2-bit errors.

Problem 2: Reliable Data Transfer (10 points)

The mistake in Figure 1 is receiver also send NAK when it receives the duplicate packet, which will lead to sender only sends the same packet, not next packet. The correct method is receiver should send ACK when it receives the duplicate packet.

Problem 3: Pipelining (20 points)

（1）Because the size of window is 4 and the next in-order packet number is k, so the sequence numbers in the window at time t is k, k+1, k+2, k+3. Assume that the sequence number range of 1,024 begins from 1, which means the minimum value of k is 1. Obviously, the maximum value of k+3 is 1024. We get two inequations and . It is easily to get the solution is . When k chooses a value belonging to [1, 1021], we get a set of sequence numbers, so there 1021 kinds of possible sets of sequence numbers.

（2）As we discuss at question 1, the possible value of k belongs to [1,1021]. Based on GBN protocol, when receiver expects packet k, it will response ACK with sequence number k-1 to sender. It is easily to get that the range of k-1 is [1,1020], so any possible value of the ACK field belongs to [1,1020].

Problem 4: TCP (20 points)

（1）

For GBN, there are 5+4=9 segments Host A has sent and 4+4=8 ACKs Host B has sent in total. The sequence numbers of ACKs are 1,1,1,1,2,3,4,5.

For SR, there are 5+1=6 segments Host A has sent and 5 ACKs Host B has sent in total. The sequence numbers of ACKs are 1,3,4,5,2.

For TCP, there are 5+1=6 segments Host A has sent and 5 ACKs Host B has sent in total. The sequence numbers of ACKs are 1,1,1,1,5.

（2）

For GBN and SR, they need to wait a long timeout to retransmit the lost packet, but TCP using Fast Retransmit does not need to wait a long timeout to retransmit the lost packet. TCP will successfully finish this task in shortest time.

Problem 5: Congestion Control (30 points)

1. When congestion window size is less than ssthresh, congestion window size will have an exponential increase. From Figure 3, we know it roughly starts at t =1st transmission round and ends at t=6th transmission round, so the intervals of time are 5 transmission rounds.
2. When congestion window size is larger than ssthresh, congestion window size will have a linear increase. From Figure 3, we know it roughly starts at t=6th transmission round and ends at t=15th transmission round, so the intervals of time are 9 transmission rounds.
3. After the 16th transmission round, congestion window size will have an linear increase, so segment loss is detected by triple duplicate ACK.
4. After the 22nd transmission round, congestion window size will have an exponential increase, so segment loss is detected by timeout.
5. Before achieving ssthresh, congestion window size will have an exponential increase at the first transmission round, so we get from Figure 2 that the initial value of ssthresh is roughly 32.
6. After that segment loss is detected by triple duplicate ACK, the congestion window size is roughly 24. Based on principle of congestion avoidance , the congestion window size is equal to ssthresh+3, so ssthresh is 21.
7. On loss event, ssthresh will be set to a half of the congestion window size before loss event. From Figure 3, we get before loss event, the congestion window size is roughly 29, so ssthresh is 14.5.
8. From 1st transmission to 6th transmission, there are roughly 63 segments sent, so the 70th segment is sent at 7th transmission.
9. When a trip duplicate event occurs, ssthresh is set to a half of cwnd before this event. Before this event, cwnd is 7, so ssthresh is roughly 3.5.
10. Based on TCP Tahoe, when triple duplicate ACKs are received at the 16th round, cwnd is set to 1, cwnd will be 7 at 19th transmission.
11. After hat triple duplicate ACKs are received at the 16th round, there 63 packets have been sent out from 17th round till 22nd round.