

# ECE 461 – Internetworking

## Problem Sheet 7

**Problem 1.** Consider the state of a sliding window at the sending side of a TCP connections as shown in Figure 1. (Each number corresponds to one byte).

- Explain the difference between the advertised window and the usable window.
- Start with the state shown in Figure 2. How many bytes can be transmitted in the shown state? What are the sequence numbers of the bytes that can be transmitted?
- Start with the state shown in Figure 2. Show how the advertised and usable windows change when the sender transmits a 2-byte long segment.
- Start with the state shown in Figure 2. Show how the advertised and usable windows change when a segment with ( $AckNo=5$ ,  $Window\ size = 5$ ) is received.
- Start with the state shown in Figure 2. Show how the advertised and usable windows change when a segment with ( $AckNo=3$ ,  $Window\ size = 5$ ) is received.

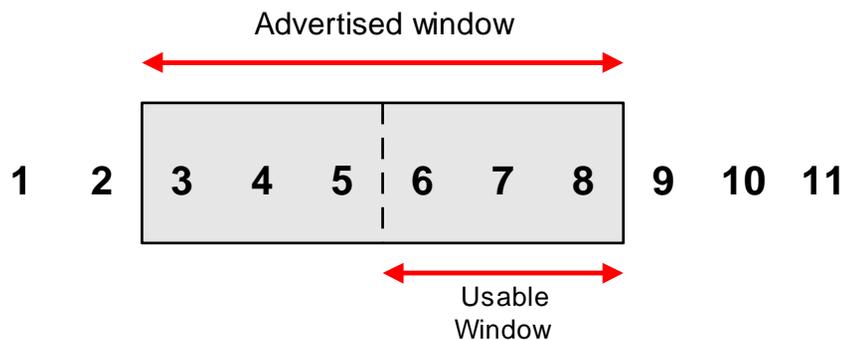
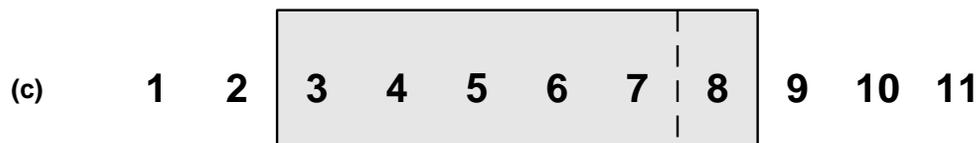
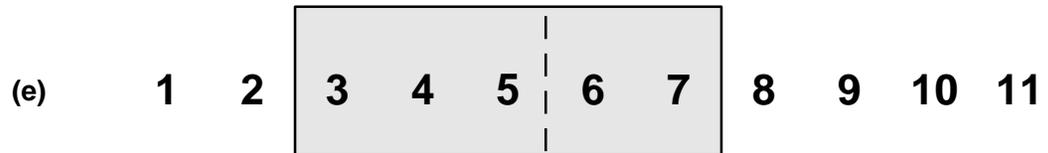
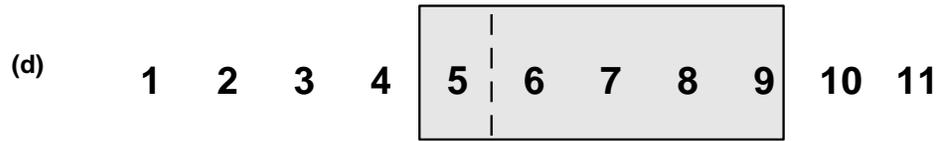


Figure 1.

- The advertised windows is the maximum number of unacknowledged bytes that the sender can transmit. The advertised window is set (=advertised) by the receiver. The usable windows is the advertised window minus the number of bytes which have been transmitted but have not been acknowledged.
- The sender can transmit 3 bytes with sequence number 6,7,8.



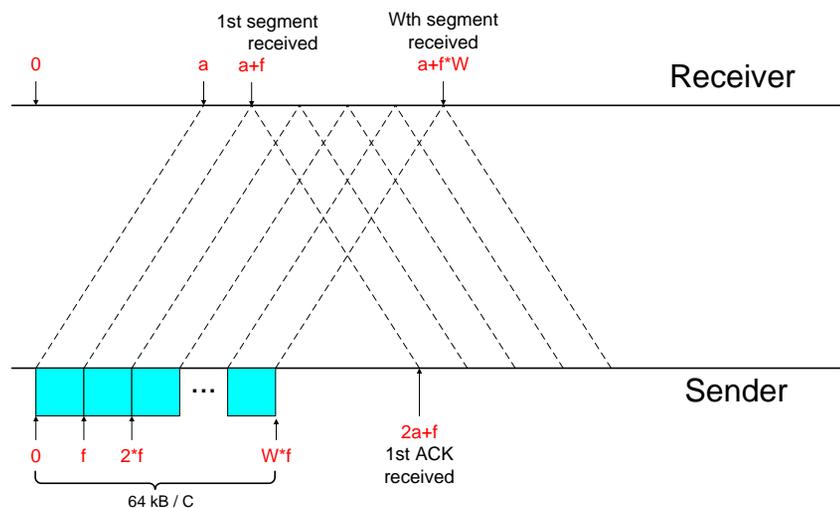


**Problem 2.** Consider a TCP connection with a roundtrip delay of 10 milliseconds. What is the maximum achievable throughput of the TCP connection? How does the throughput change if the roundtrip time increases to 40 milliseconds?

**Solution:**

This problem assumes material from ECE 361 on the throughput of the sliding window flow control protocol.

- The maximum throughput happens when the window size has maximal size. The figure shows the transmission scenario.



We use parameters:

$C$  = capacity of network (in bit/sec)

$f$  = transmission time of a segment

We assume that the size of all segments is  $MSS$  bytes. Then  $f = \frac{MSS \times 8}{C}$  (in seconds).

Note that we must multiply by 8 since "1 byte = 8 bits".

$a$  = propagation delay between sender and receiver (roundtrip delay =  $2a$ )

$W$  = maximum window size in segments of size  $MSS$  bytes

The sender transmits  $W$  segments, each with maximum size. The receiver sends ACKs to the sender (We assume that the transmission time of an ACK is negligible). It takes  $Wf$  time to transmit the entire window. The first ACK arrives at the sender at time  $2a+f$ .

There are two scenarios:

**Scenario 1:  $2a + f \leq Wf$**

Here, the first acknowledgement arrives before the entire window (with  $W$  segments) has been transmitted. In this case, the sender can transmit continuously. Here the throughput (TPut) is equal to the capacity  $C$ :

$$\text{Maximum Throughput} = C$$

Using  $f = MSS/C$ , we get from  $2a + f \leq Wf$  that

$$C \leq \frac{MSS \times 8 \times (W - 1)}{2a}$$

To compute numbers we need to insert a number of assumptions.

- We assume that the maximum size of the flow control window is 64 KB ( $2^{16}$ ).
- We assume that  $MSS = 1k$  (1024 bytes). Then,  $W = 64$ .

For  $2a=10ms$ , we get:  $C \leq 51.6$  Mbps

For  $2a = 40ms$ , we get:  $C \leq 12.9$  Mbps

**Scenario 2:  $2a + f > Wf$**

In this case, the entire window (of 64kB) has been transmitted before the first acknowledgement has arrived. Here, the sender has to wait for the acknowledgment to arrive, before it can send more data.

Therefore, the sender can transmit at most  $W$  segments in each time interval of  $2a + f$ . Note that  $W$  segments corresponds to  $W \times 8 \times MSS$  bits. So, the throughput is

$$\text{Maximum Throughput} = \frac{W \times 8 \times MSS}{2a + f} = \frac{W \times 8 \times MSS}{2a + \frac{MSS \times 8}{C}}$$

Let us consider a fast network, where  $C \rightarrow \infty$ , then the formula simplifies to

$$\text{Maximum Throughput} \approx \frac{W \times 8 \times MSS}{2a}$$

For  $2a=10ms$ , we get:  $C = 52.4$  Mbps

For  $2a = 40\text{ms}$ , we get:  $C = 13.1 \text{ Mbps}$

In both scenarios, we roughly obtain that

$$\text{Maximum Throughput} = \frac{\text{max. window size (in bits)}}{\text{round trip delay}}$$

Therefore, the maximum throughput decreases proportionally to the round trip delay.

**Problem 3.** Assume that we have a TCP connection between A and B. Assume that A uses slow start and congestion avoidance with the following initial values:

Congestion window (at time=0):  $cwnd = 12$  segment.

Slow-start threshold:  $ssthresh = 5$  segments.

For the purposes of this problem, assume that  $MSS = 100$  Bytes.

Assume that the following events occur at A:

Time  $t = 0$ : A sends segment with 100 bytes to B, starting with  $SeqNo = 0$ .

Time  $t = 1$ : A receives an ACK with  $AckNo = 100$

Time  $t = 2$ : A sends segment with 100 bytes to B, starting with  $SeqNo = 100$ .

Time  $t = 3$ : A sends segment with 100 bytes to B, starting with  $SeqNo = 200$ .

Time  $t = 4$ : A receives an ACK with  $AckNo = 100$ .

Time  $t = 5$ : A sends segment with 100 bytes to B, starting with  $SeqNo = 300$ .

Time  $t = 6$ : A receives an ACK with  $AckNo = 100$ .

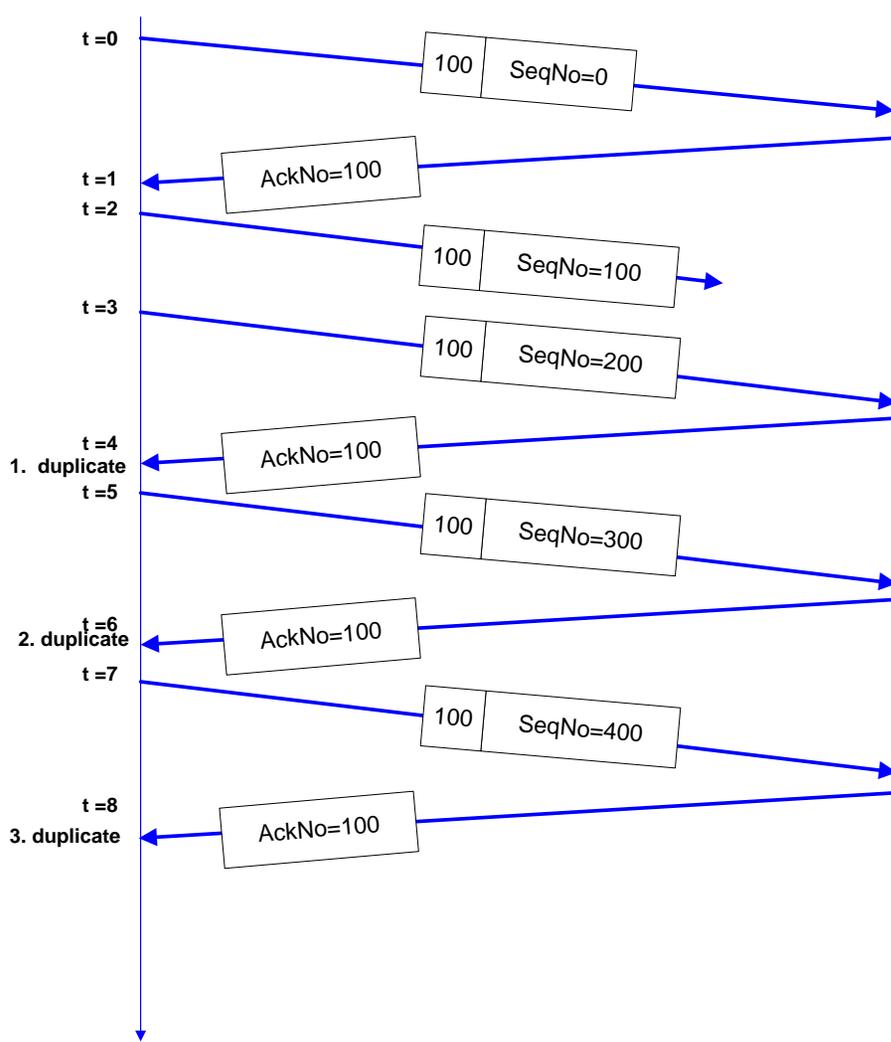
Time  $t = 7$ : A sends segment with 100 bytes to B, starting with  $SeqNo = 400$ .

Time  $t = 8$ : A receives an ACK with  $AckNo = 100$ .

- Describe the actions performed by TCP Tahoe at time  $t=8$ , and describe the values of  $cwnd$  and  $ssthresh$  after the actions are performed.
- Describe the actions performed by TCP Reno at time  $t=8$ , and describe the values of  $cwnd$  and  $ssthresh$  after the actions are performed.
- For both TCP Tahoe and TCP Reno, describe the actions performed when a timeout occurs between times  $t=5$  and  $t=6$ .

*Solutions:*

*The figure below shows the transmission scenario.*



(a) After the 3<sup>rd</sup> duplicate ACK, TCP Tahoe performs Fast Retransmit and enters the slow start phase:

$$\begin{aligned} \text{ssthresh} &= \text{cwnd}/2 \\ \text{cwnd} &= 1 \end{aligned}$$

At t=0: cwnd = 12, ssthresh = 5 (→ we are in the congestion avoidance phase)

At t=1: cwnd =  $12 \frac{1}{12}$ , ssthresh = 5

No changes to cwnd at t=4 and t=6 since no new data is acknowledged.

At t=8: ssthresh =  $6 \frac{1}{24}$  (fractional part can be ignored since we are counting segments)

$$\text{cwnd} = 1$$

(b) After the 3<sup>rd</sup> duplicate ACK, TCP Reno performs Fast Retransmit/Fast Recovery:

- $ssthresh = cwnd/2$
- $cwnd = ssthresh + 3$

At  $t=0$ :  $cwnd = 12$ ,  $ssthresh = 5$  ( $\rightarrow$  we are in the congestion avoidance phase)

At  $t=1$ :  $cwnd = 12 \frac{1}{12}$   $ssthresh = 5$

No changes to  $cwnd$  at  $t=4$  and  $t=6$  since no new data is acknowledged.

At  $t=8$ :  $ssthresh = 6 \frac{1}{24}$  (fractional part can be ignored since we are counting segments)

$cwnd = 9 \frac{1}{24}$

c) When retransmission timeout happens between  $t=5$  and  $t=6$ , enter the slow start phase:

$ssthresh = cwnd/2 = 6 \frac{1}{24}$

$cwnd = 1$