COMPUTER COMMUNICATION NETWORKS

Department of Electronics and Communication Engineering

COMPUTER COMMUNICATION NETWORKS

Unit 2 - Numerical

Dr. Arpita Thakre

Department of Electronics and Communication Engineering

(1) Consider Stop & Wait Protocol. Host A is sending packets to host B where round trip time is 30 msec. The host A to host B link supports data transmission at 1 Gbps. Suppose the size of a packet is 1500 bytes. Calculate the link utilization.

Ans : RTT+L/R = $\{30 + (1500*8/10^6)\}$ = 30.012 msec

30.012 msec is needed to reliably send 1 packet from host A to host B, assuming transmission delay for ACK (from B to A) is negligible.

Therefore link utilization is 0.012 msec/(30.012 msec) = 0.00039

(2) Consider Pipelining Protocol that uses a window of size N. Host A is sending packets to host B where round trip time is 30 msec. The host A to host B link supports data transmission at 1 Gbps. Suppose the size of a packet is 1500 bytes. What should be N such that the link utilization becomes 0.98 ?

Ans : $\{30 + (1500*8/10^6)\}$ = 30.012 msec is needed to reliably send 1 packet from host A to host B, assuming transmission delay for ACK (from B to A) is negligible. However within 30.012 msec, N packets are already out in the channel. Transmission delay of N packets is N*0.012 msec.

Therefore, link utilization = (N*0.012) /(30+ 0.012) = 0.98

i.e., N = (.98*30.012)/(0.012) = 2450.98 = 2451

(**3**) Consider the GBN protocol with a sender window size of 4 and a sequence number range of 1,024. Suppose that at time t, the next in-order packet that the receiver is expecting has a sequence number of k. Assume that the medium does not reorder messages. Answer the following questions:

a. What are the possible sets of sequence numbers inside the sender's window at time t? Justify your answer.

b. What are all possible values of the ACK field in all possible messages currently propagating back to the sender at time t? Justify your answer.

(**3a)** We have a window size of 4. Suppose the receiver has received packet k-1, and has ACKed all other preceding packets. If all of these ACK's have been received by sender, then sender's window is [k,k+1, k+2, k+3].

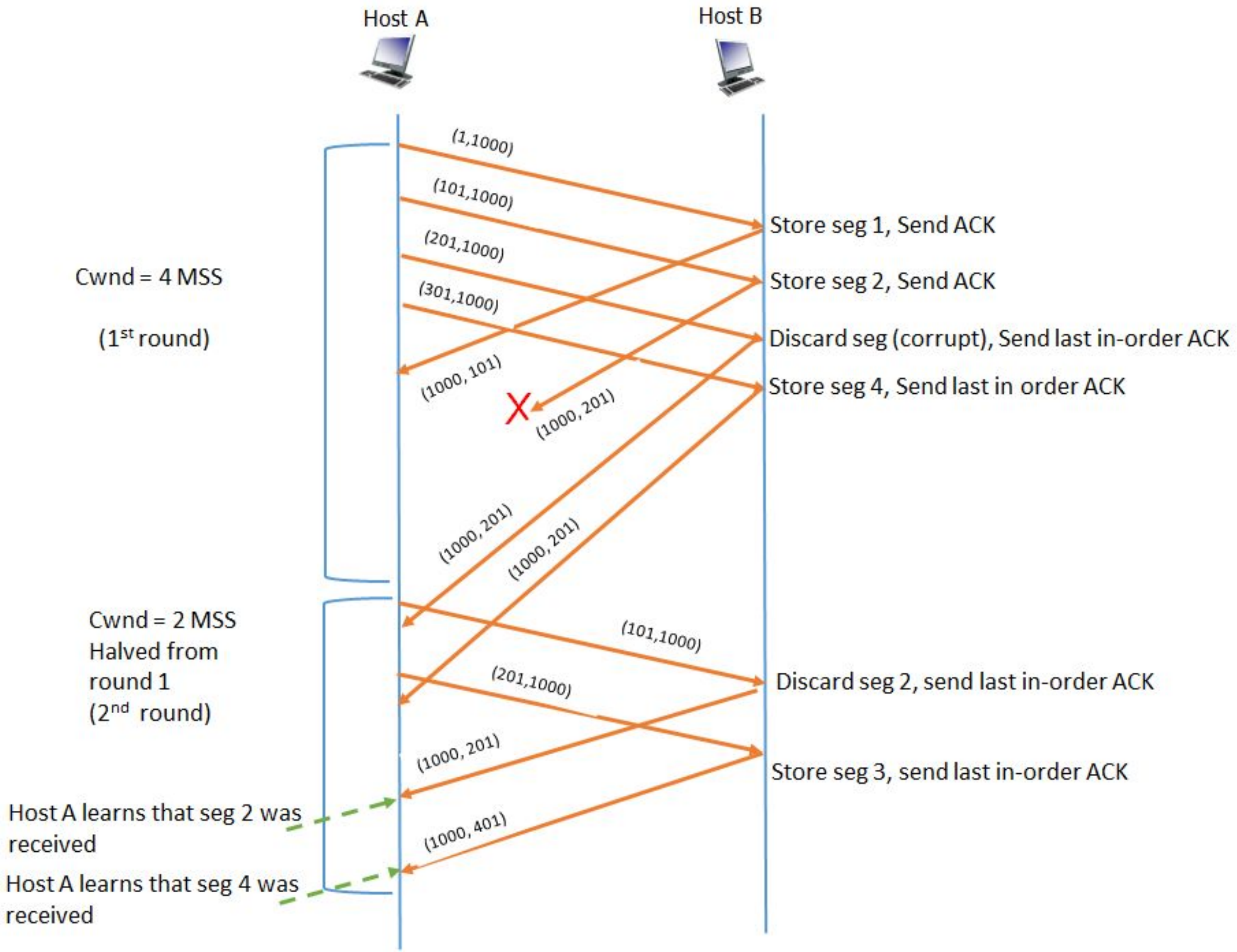
Suppose next that none of the ACKs have been received by the sender. In this second case, the sender's window contains k-1 and the N packets up to and including k-1. The sender's window is thus [k-4,k-3,k-2,k-1].

(**3b**) If the receiver is waiting for packet k, then it has received (and ACKed) packet k-1 and the N-1 packets before that. If none of those N ACKs have been yet received by the sender, then ACK messages with values of [k-N,k-1] may still be propagating back. Because the sender has sent packets [k-N, k-1], it must be the case that the sender has already received an ACK for k-N-1. Once the receiver has sent an ACK for packet k-N-1 it will never send an ACK that is less that k-N-1. Thus the range of in-flight ACK values can range from k-N-1 to k-1.

(4) Consider 4 packets, each having 100 bytes, are transmitted by host A to host B using TCP with only congestion avoidance. Assume TCP connection was already complete. Let the sequence numbers of host A and host B to be 1 and 1000 respectively, for subsequent communication.

- Segment 3 corrupt and ACK 2 Lost
- Suppose both the ACKs corresponding to segments 3 and 4 **arrive after timeout.**

Draw the timing diagram that depicts sequence number and acknowledgement number for each packet transaction.

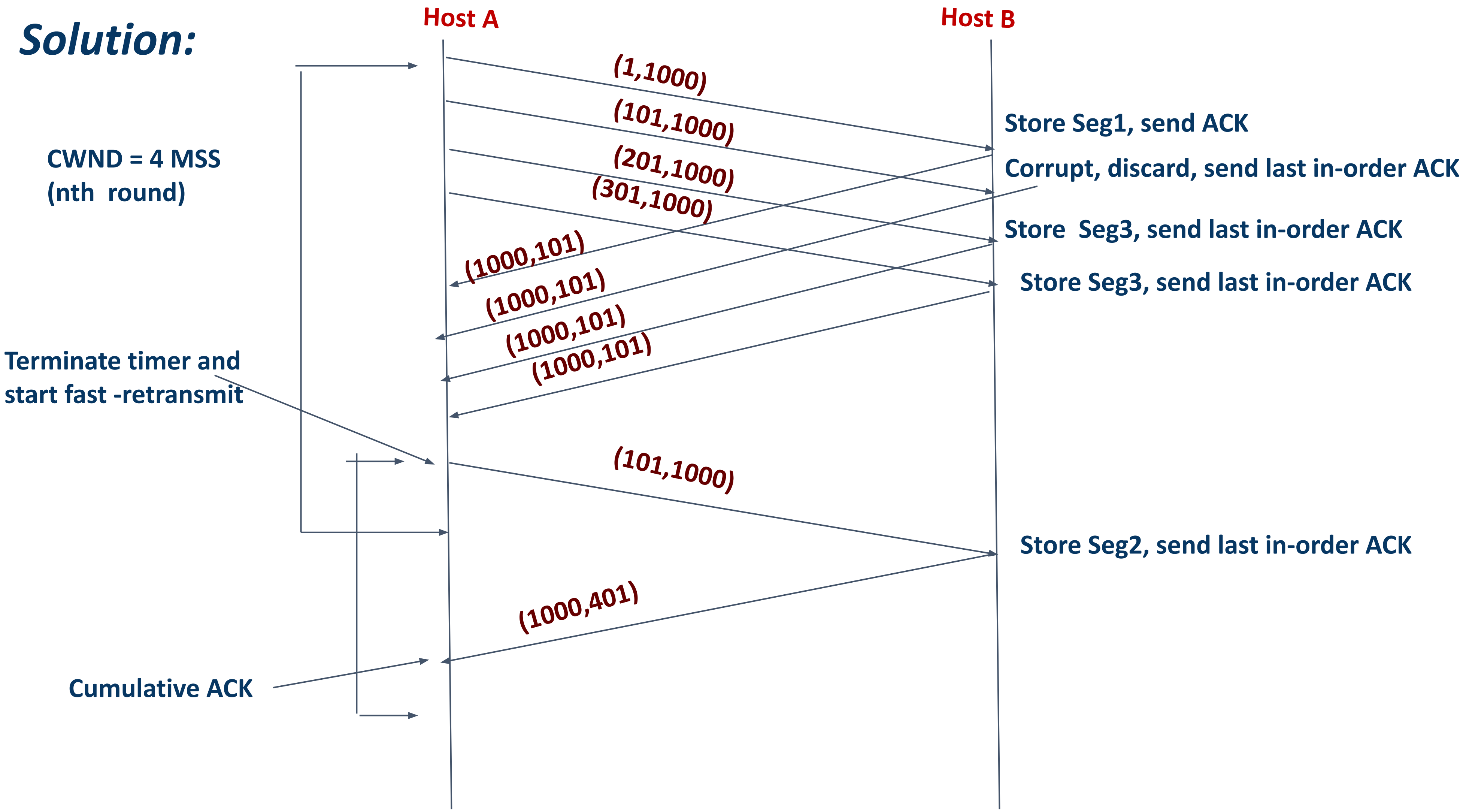


Host A
Host B
(1,1000)
(101,1000)
(201,1000)
(301,1000)
Store seg 1, Send ACK
Store seg 2, Send ACK
Discard seg (corrupt), Send last in-order ACK
Store seg 4, Send last in order ACK
Cwnd = 4 MSS
(1st round)
(1000, 101)
(1000, 201)
(1000, 201)
(1000, 201)
Cwnd = 2 MSS
Halved from
round 1
(2nd round)
(101,1000)
(201,1000)
Discard seg 2, send last in-order ACK
Store seg 3, send last in-order ACK
(1000, 201)
(1000, 401)
Host A learns that seg 2 was
received
Host A learns that seg 4 was
received

(5) Consider 4 packets, each having 100 bytes, are transmitted by host A to host B using **TCP Reno**. Assume TCP connection was already complete. Let the sequence numbers of host A and host B to be 1 and 1000 respectively, for subsequent communication. Sequence numbers from B to A can be ignored.

- Suppose second packet is corrupted during transmission but others are received correctly by host B. Draw a timing diagram including the retransmissions. Assume retransmissions are successful. Timing diagram must depict sequence number and acknowledgement number for each packet transaction.

Solution:

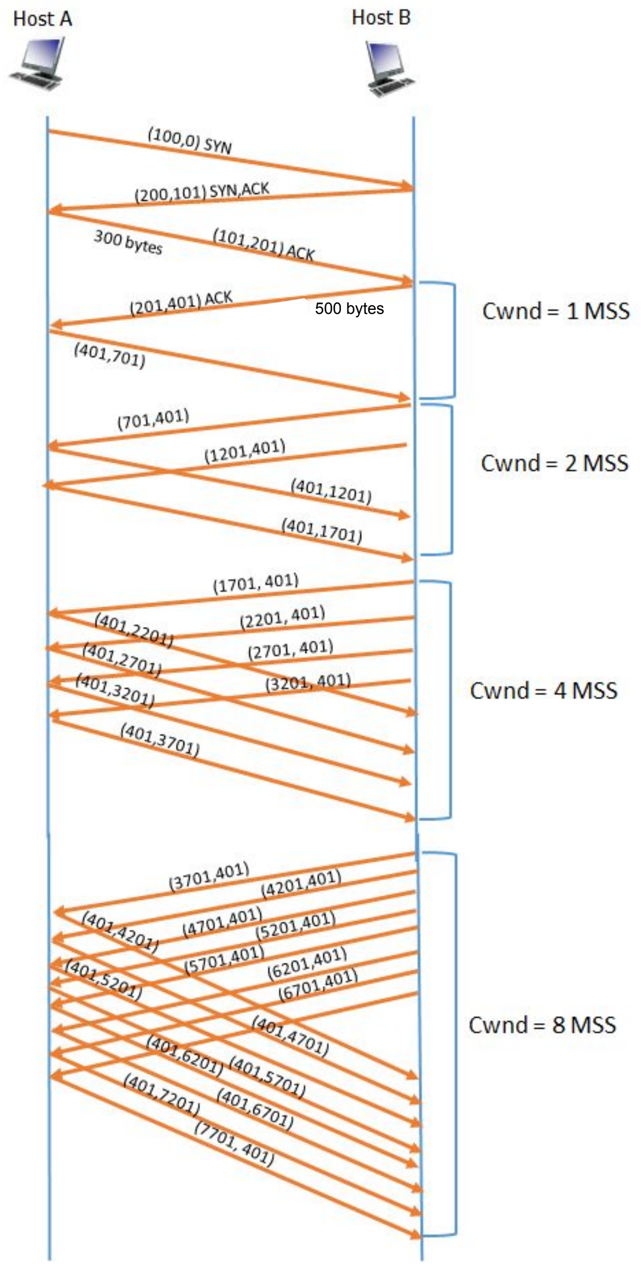

Host A
Host B
CWND = 4 MSS
(nth round)
(1,1000)
(101,1000)
(201,1000)
(301,1000)
Store Seg1, send ACK
Corrupt, discard, send last in-order ACK
Store Seg3, send last in-order ACK
Store Seg3, send last in-order ACK
(1000,101)
(1000,101)
(1000,101)
(1000,101)
Terminate timer and
start fast -retransmit
(101,1000)
Store Seg2, send last in-order ACK
(1000,401)
Cumulative ACK

(6) : Let 100 and 200 be the Sequence numbers of Hosts A and B at the start of the TCP connection. Let A send a HTTP request of 300 bytes and let B reply the object in 15 segments of 500 bytes each. Let all transmissions be successful. Let MSS= 500 bytes. Show the timing diagram under TCP (TCP Reno / Taho).

Case1: No congestion
Case 2: Maximum number of segments = 5 MSS.

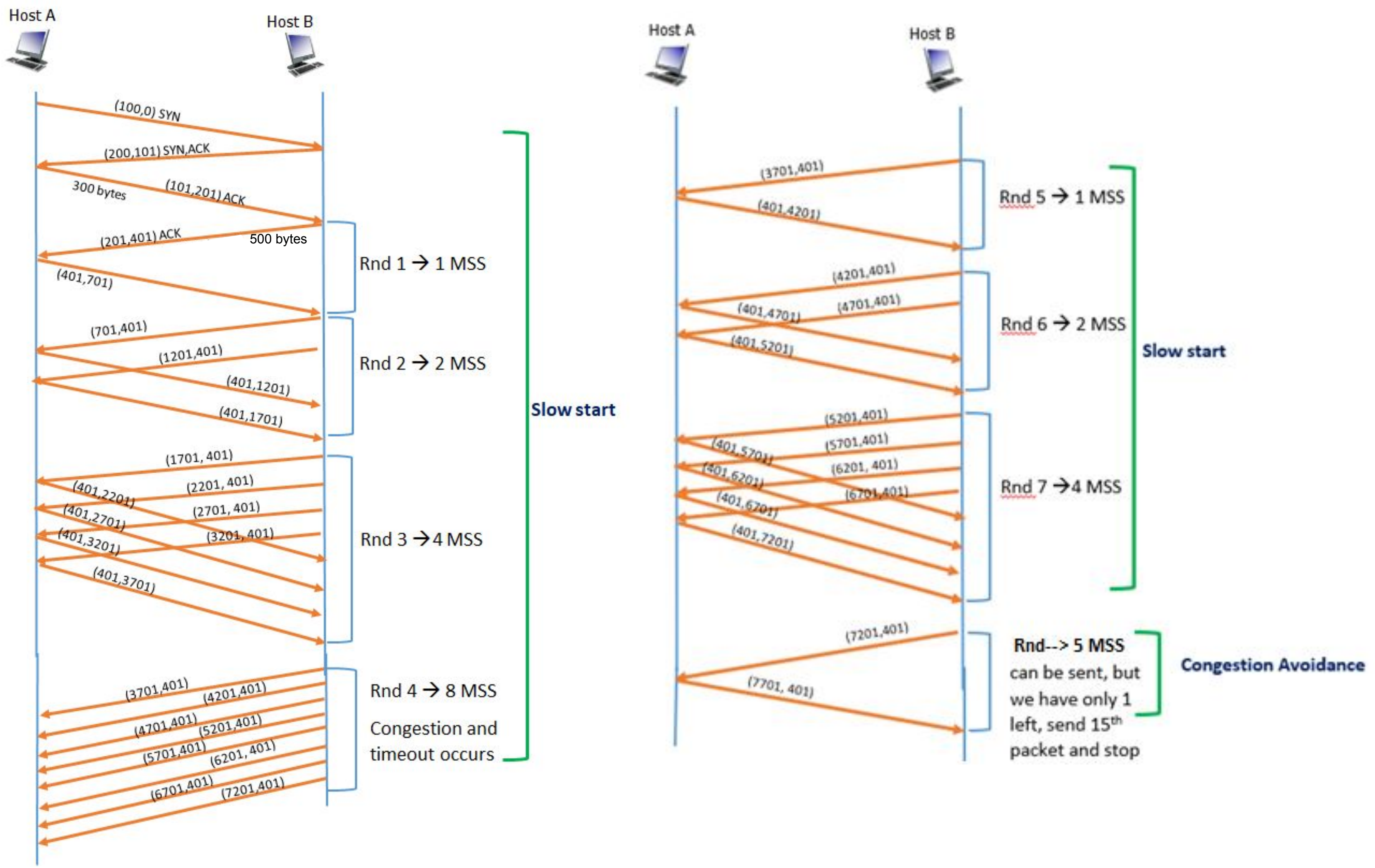
Solution for Case 1:

No Congestion



Solution for Case 2:

Maximum number of segments = 5 MSS.



(7) Consider transferring an enormous file of *L* bytes from Host A to Host B. Assume an MSS of 536 bytes.

a. What is the maximum value of *L* such that TCP sequence numbers are not exhausted? Recall that the TCP sequence number field has 4 bytes.

b. For the *L* you obtain in (a), find how long it takes to transmit the file. Assume that a total of 66 bytes of transport, network, and data-link header are added to each segment before the resulting packet is sent out over a 155 Mbps link. Ignore flow control and congestion control so A can pump out the segments back to back and continuously

There are $2^{32} = 4{,}294{,}967{,}296$ possible sequence numbers.

a) The sequence number does not increment by one with each segment. Rather, it increments by the number of bytes of data sent. So the size of the MSS is irrelevant -- the maximum size file that can be sent from A to B is simply the number of bytes representable by $2^{32} \approx 4.19$ Gbytes

b) The number of segments is $\left\lceil \frac{2^{32}}{536} \right\rceil = 8{,}012{,}999$. 66 bytes of header get added to each segment giving a total of 528,857,934 bytes of header. The total number of bytes transmitted is $2^{32} + 528{,}857{,}934 = 4.824 \times 10^9$ bytes.

Thus it would take 249 seconds to transmit the file over a 155~Mbps link.

THANK YOU

Dr. Arpita Thakre

Department of Electronics and Communication Engineering