COMPUTER COMMUNICATION NETWORKS

Department of Electronics and Communication Engineering

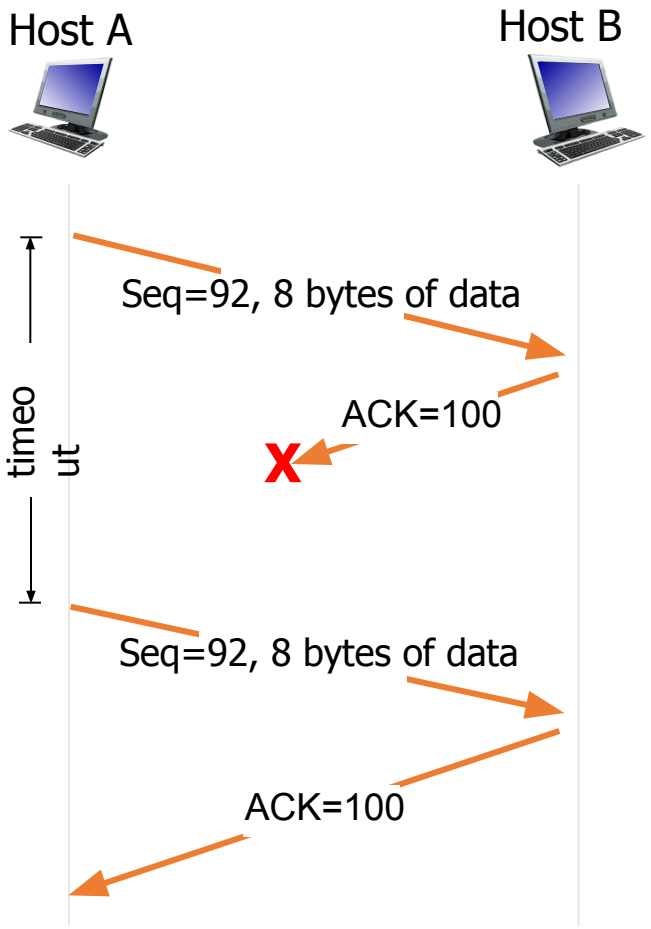
COMPUTER COMMUNICATION NETWORKS

TCP : Duplicate ACKs & Retransmissions
Flow Control

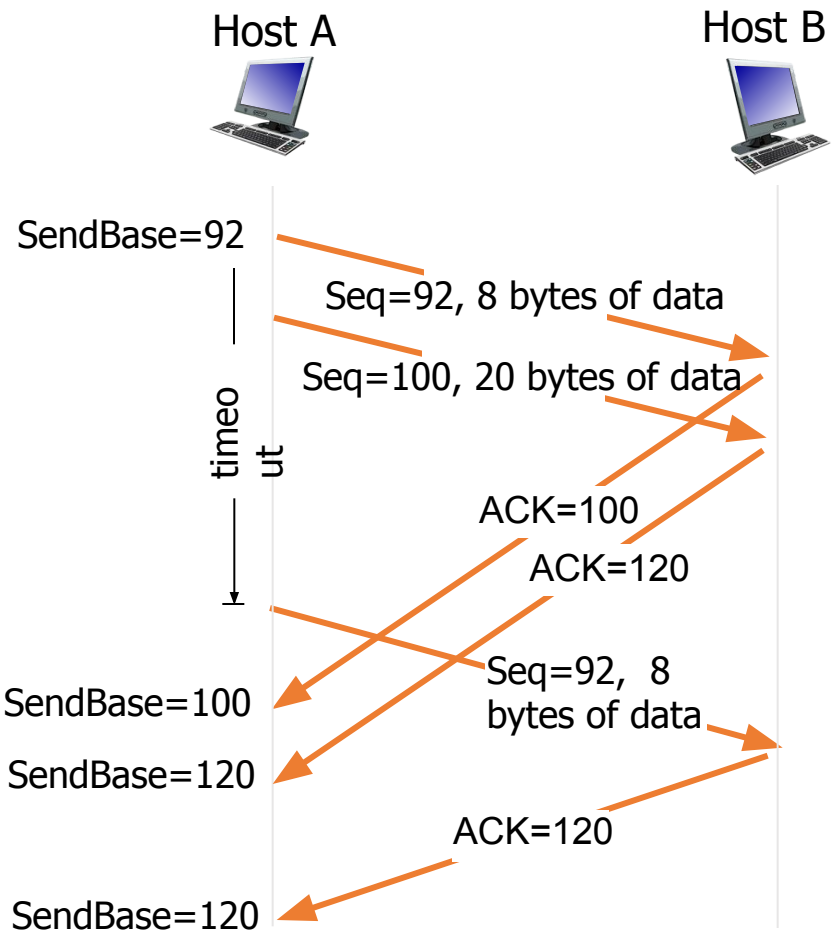
Dr. Arpita Thakre
Department of Electronics and Communication Engineering

COMPUTER COMMUNICATION NETWORKS

TCP Retransmission Scenarios



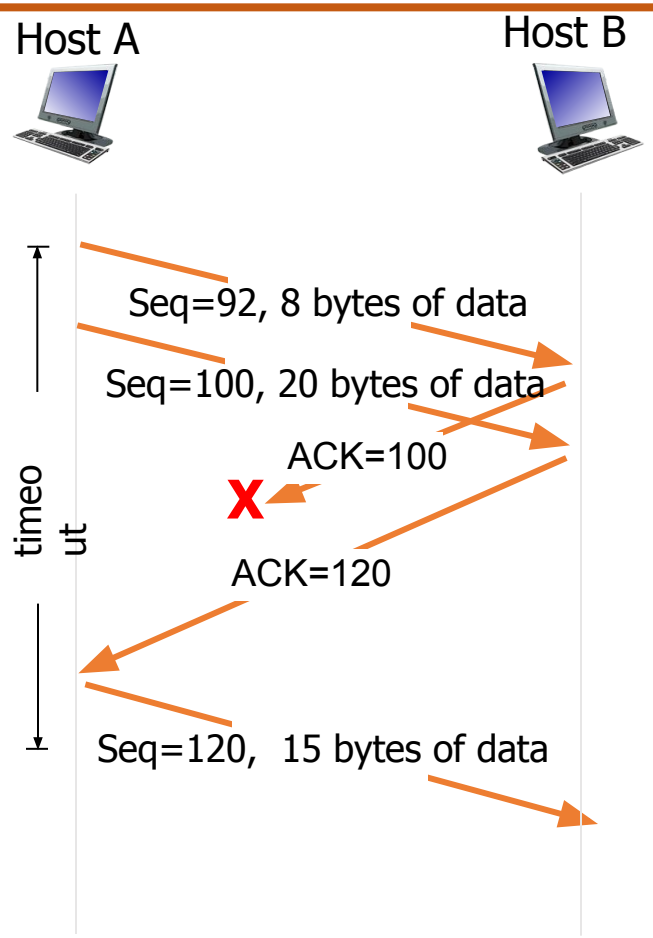
lost ACK scenario



premature timeout

COMPUTER COMMUNICATION NETWORKS

TCP Retransmission Scenarios



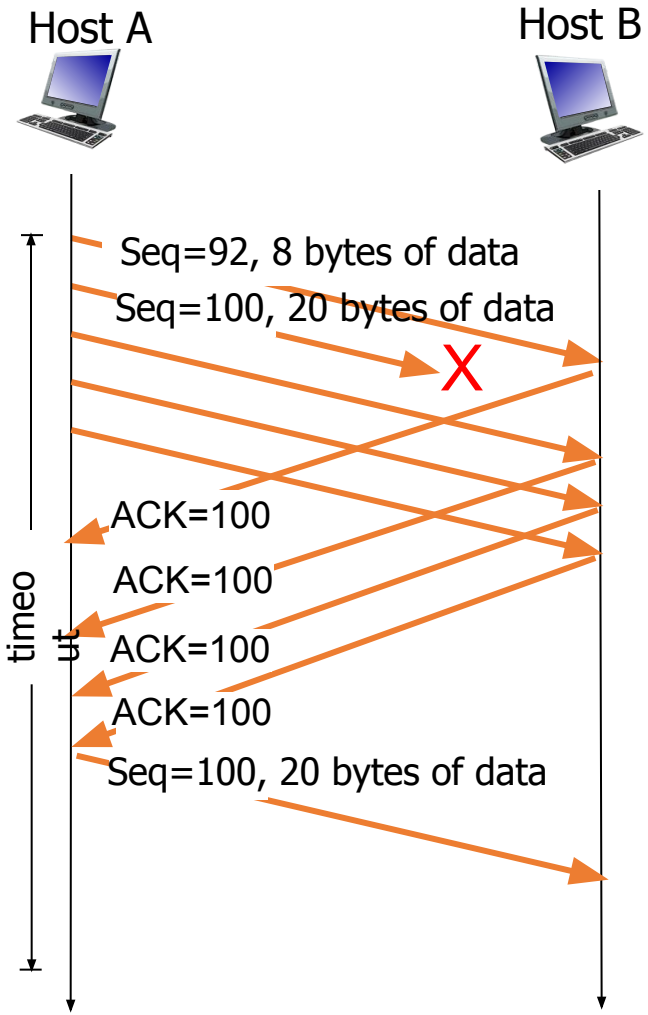
cumulative ACK

TCP ACK Generation

event at receiver	*TCP receiver action*
arrival of in-order segment with expected seq #. All data up to expected seq # already ACKed	delayed ACK. Wait up to 500ms for next segment. If no next segment, send ACK
arrival of in-order segment with expected seq #. One other segment has ACK pending	immediately send single cumulative ACK, ACKing both in-order segments
arrival of out-of-order segment higher-than-expect seq. # . Gap detected	immediately send *duplicate ACK*, indicating seq. # of next expected byte
arrival of segment that partially or completely fills gap	immediate send ACK, provided that segment starts at lower end of gap

TCP Fast Retransmit

TCP fast retransmit



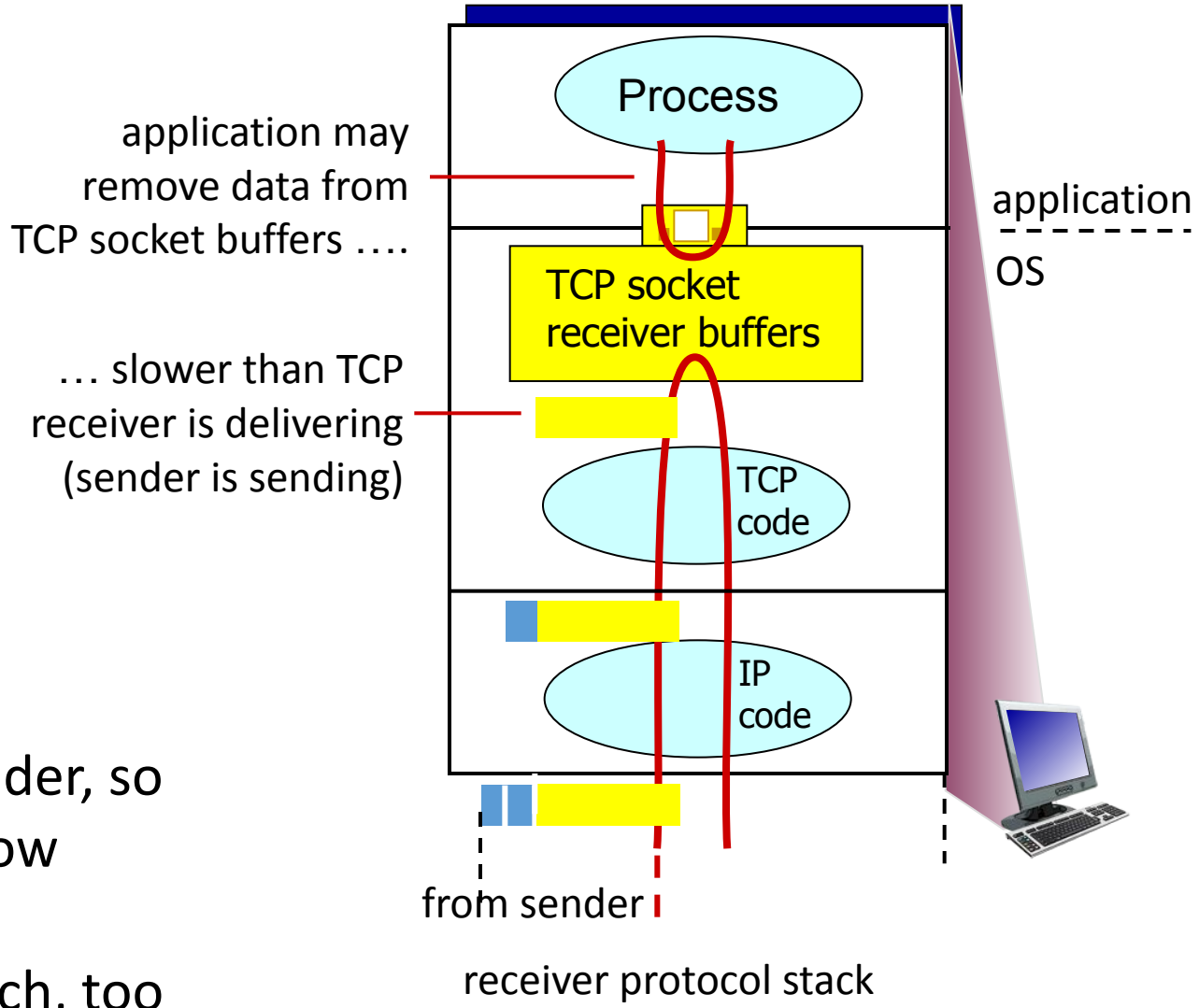
fast retransmit after sender receipt of triple duplicate ACK

TCP Flow control

- TCP provides a flow-control service to its applications to eliminate the possibility of sender overflowing the receiver's buffer.
- Flow control is thus a speed-matching service – matching the rate at which sender is sending against the rate at which receiver application is reading.
- UDP does not provide flow control

TCP Flow control

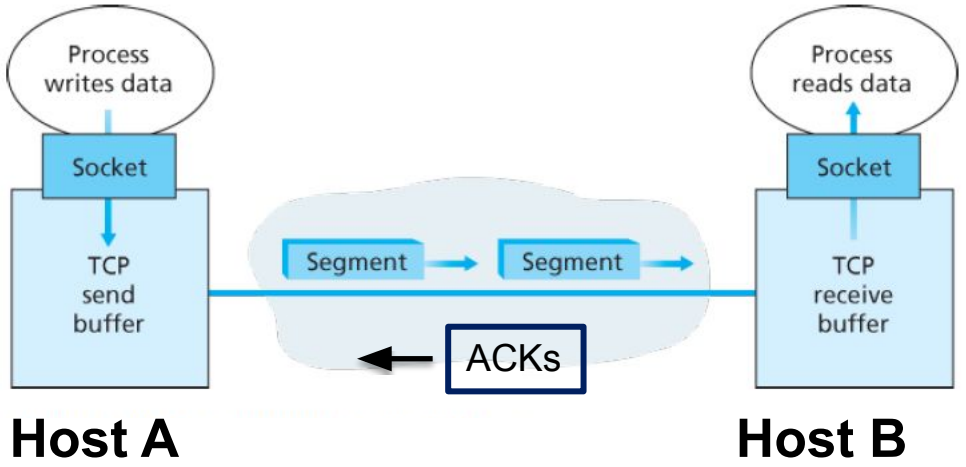
Flow Control

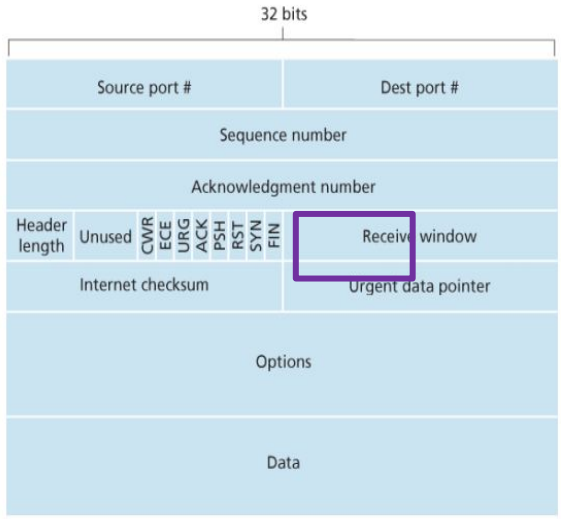


receiver informs sender, so sender won't overflow receiver's buffer by transmitting too much, too fast

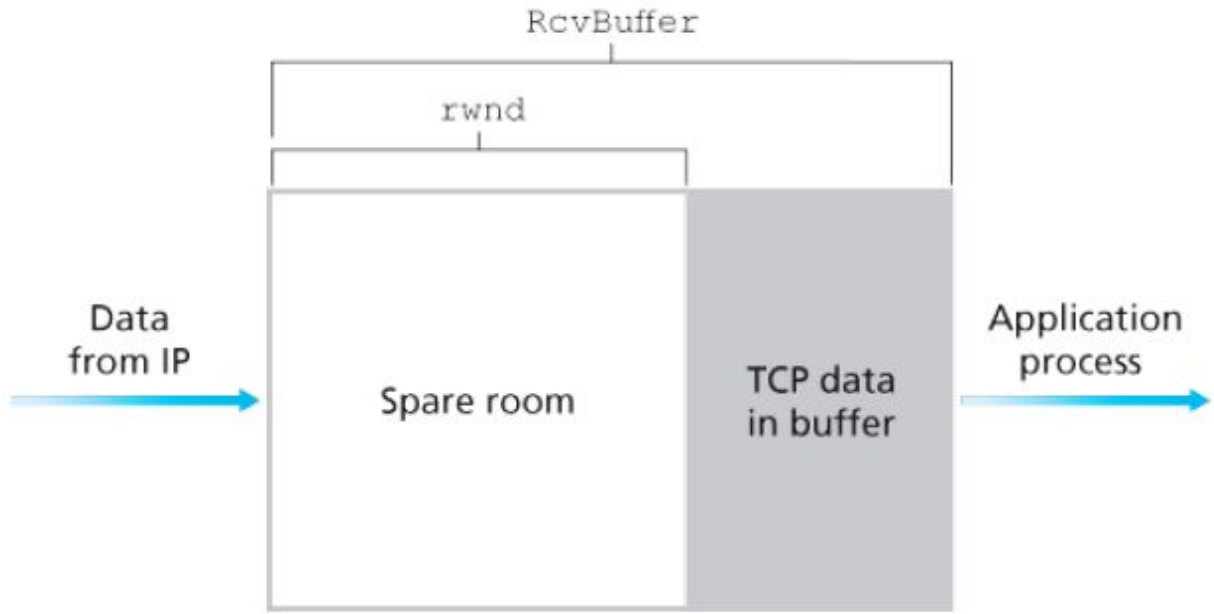
COMPUTER COMMUNICATION NETWORKS
TCP Flow control

Flow Control (contd.)



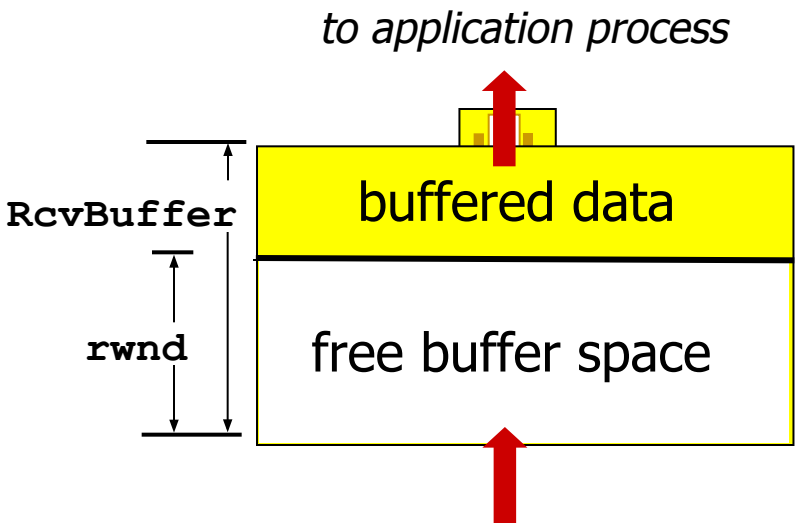


TCP segment



Flow Control (contd.)

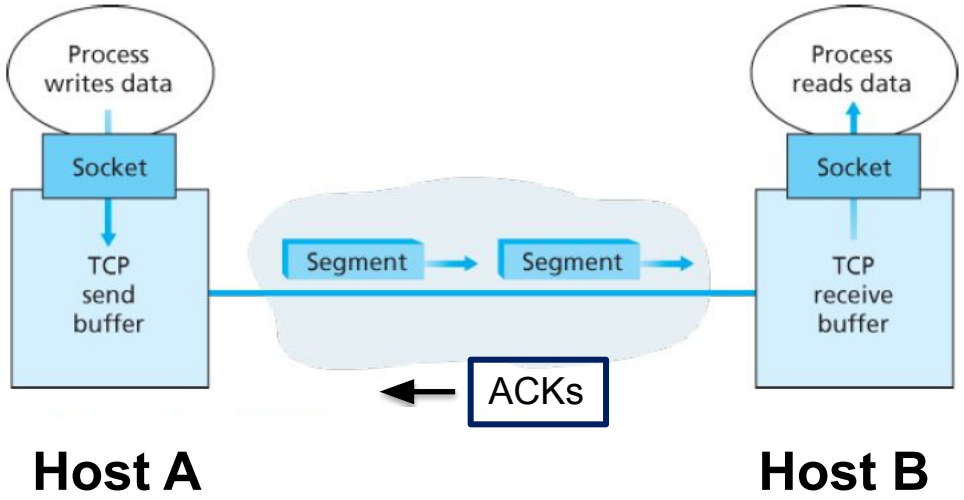
- receiver "advertises" free buffer space by including **rwnd** value in TCP header
  - **RcvBuffer** size set via socket options (typical default is 4096 bytes)
  - many operating systems autoadjust **RcvBuffer**
- sender limits amount of unACKed ("in-flight") data to receiver's **rwnd** value
- this is how sender guarantees receive buffer will not overflow
- **rwnd** is dynamic; it can be different at different times



receiver-side buffering

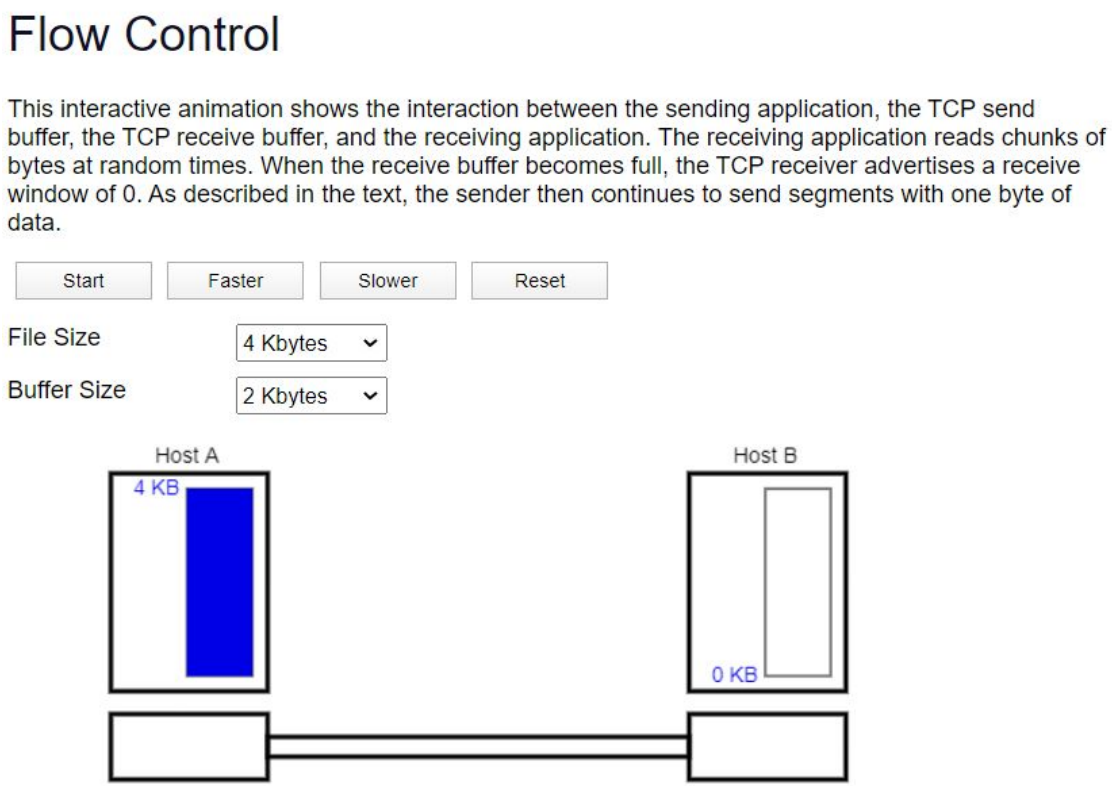
Special case

- Host B advertises **rwnd** = 0 to host A & Host B has nothing to send to host A -> host B will not be able advertise **rwnd** value to host A even if **rwnd** changes after some time.
- Consequence: host A cannot send further packets to host B, host A is blocked !
- Solution: TCP protocol mandates that host A send segment of 1 data byte to host B even when **rwnd** = 0, and host B acknowledge this segment
- **rwnd** at host B will increase as time progresses and ACK from host B will now contain updated **rwnd** > 0



COMPUTER COMMUNICATION NETWORKS
TCP Flow control

Animation Link : https://media.pearsoncmg.com/aw/ecs_kurose_compnetwork_7/cw/content/interactiveanimations/flow-control/index.html



THANK YOU

Dr. Arpita Thakre

Department of Electronics and Communication Engineering