Computer Communication Networks

Transport Layer

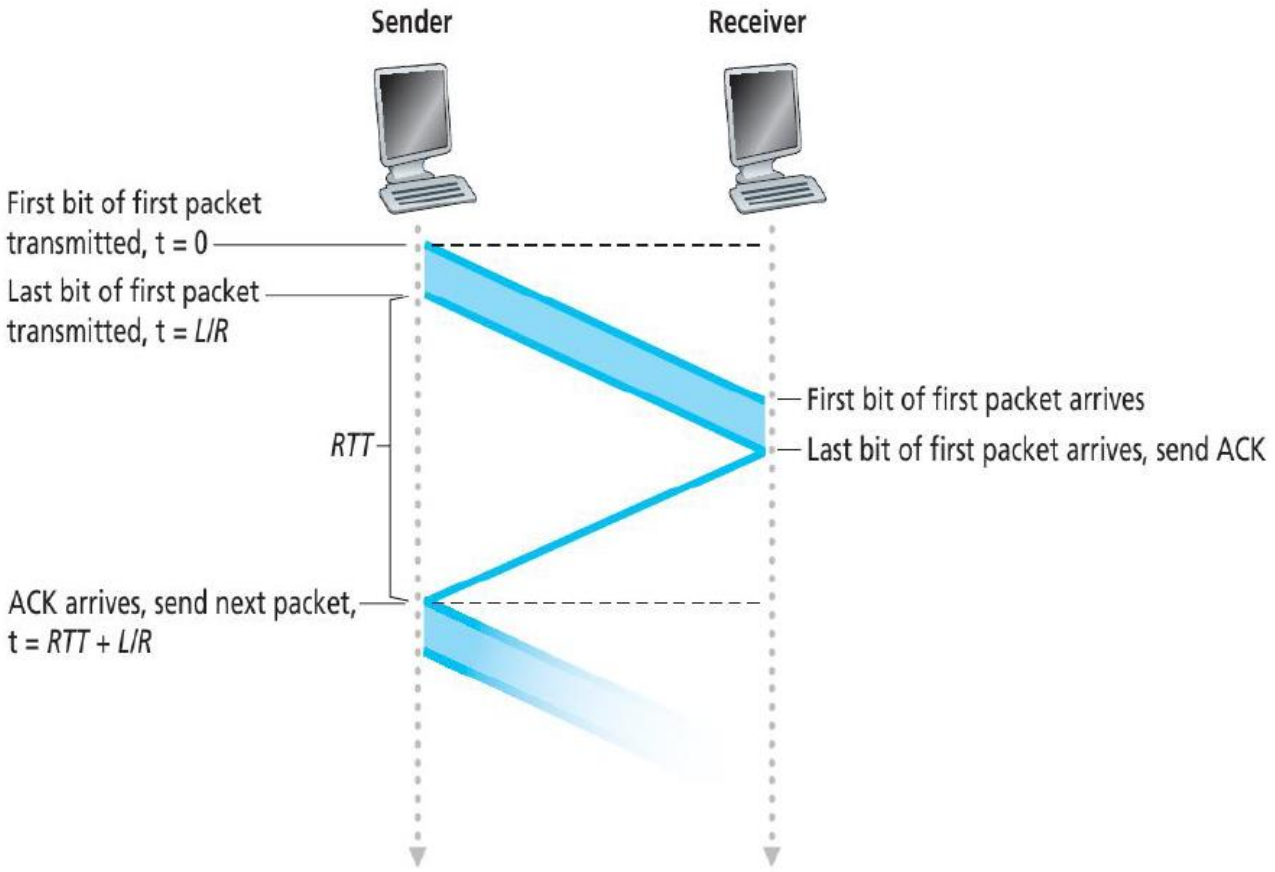
Dr. Raja Vara Prasad

Assistant Professor

IIIT Sri City

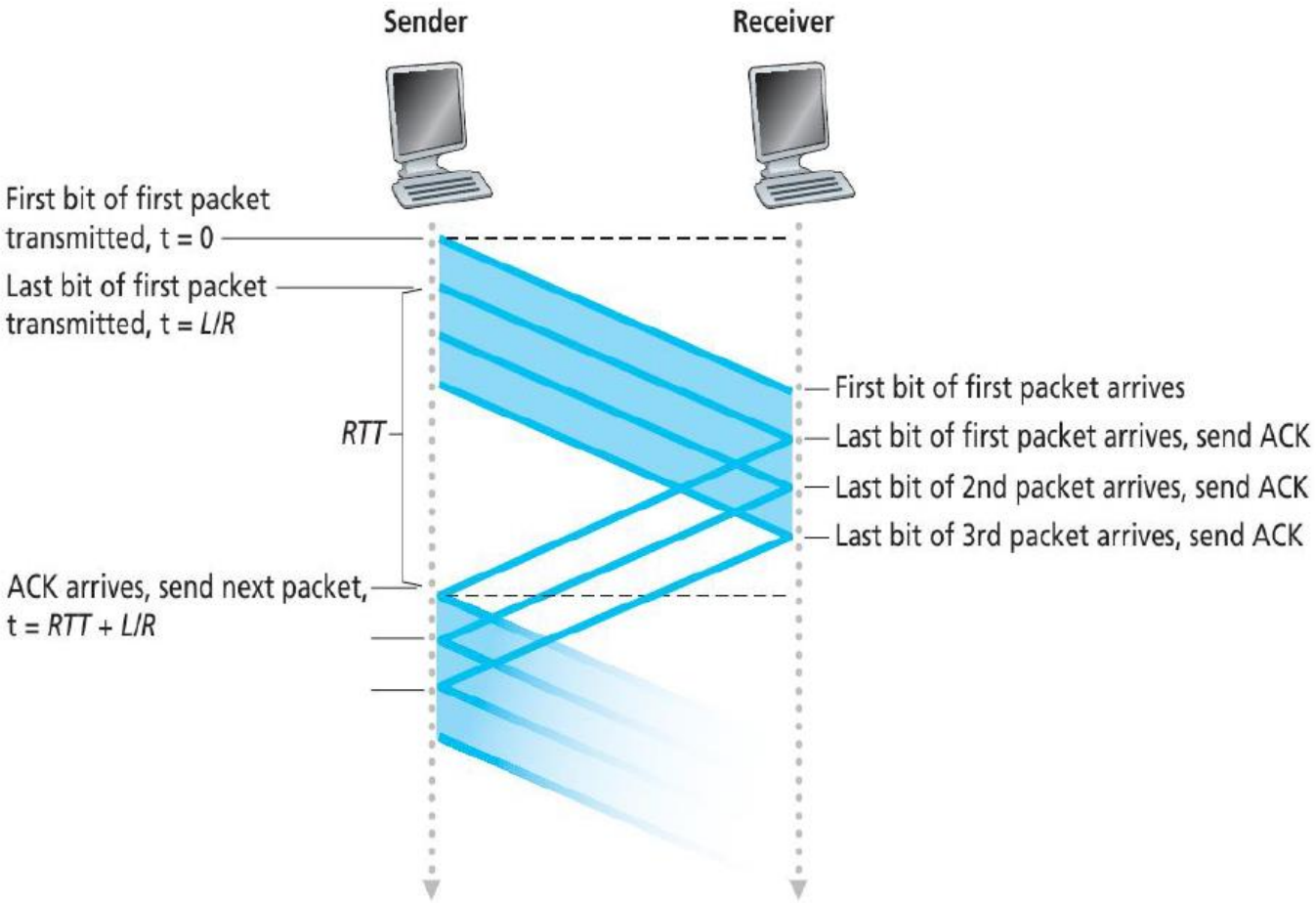
Transport Layer

Stop-and-Wait Operation



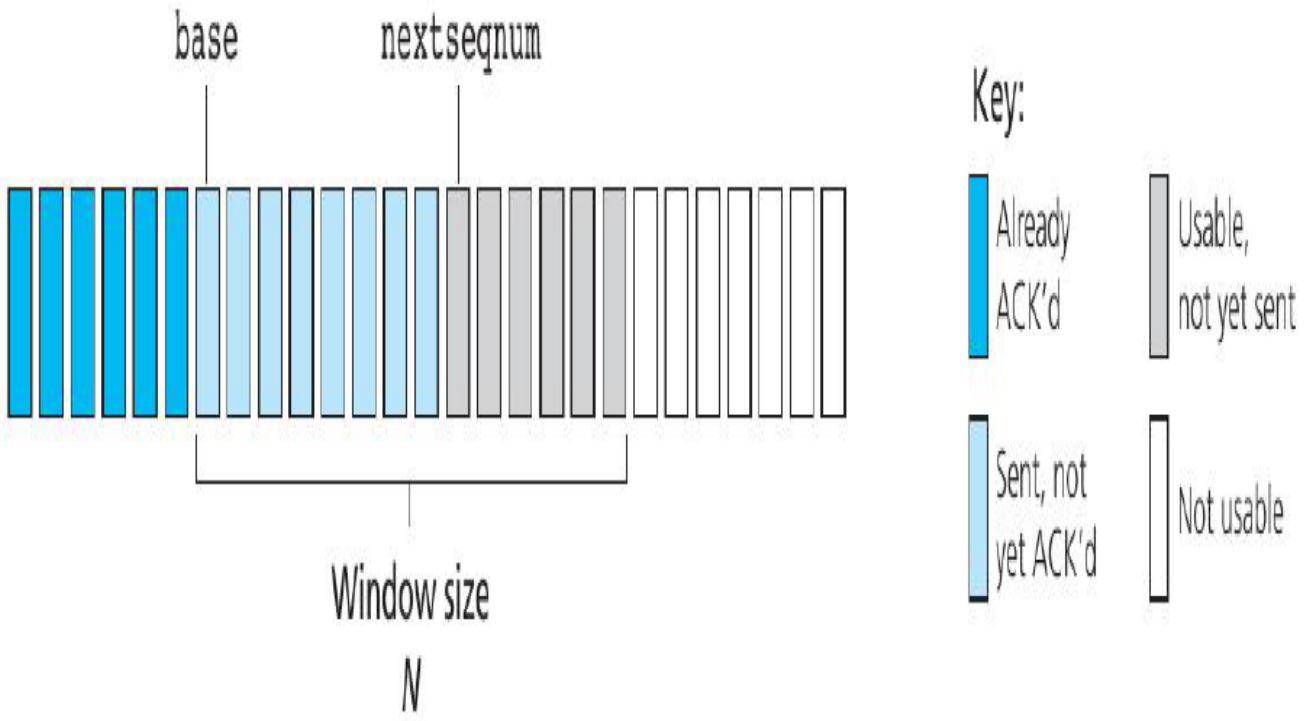
a. Stop-and-wait operation

Pipelining

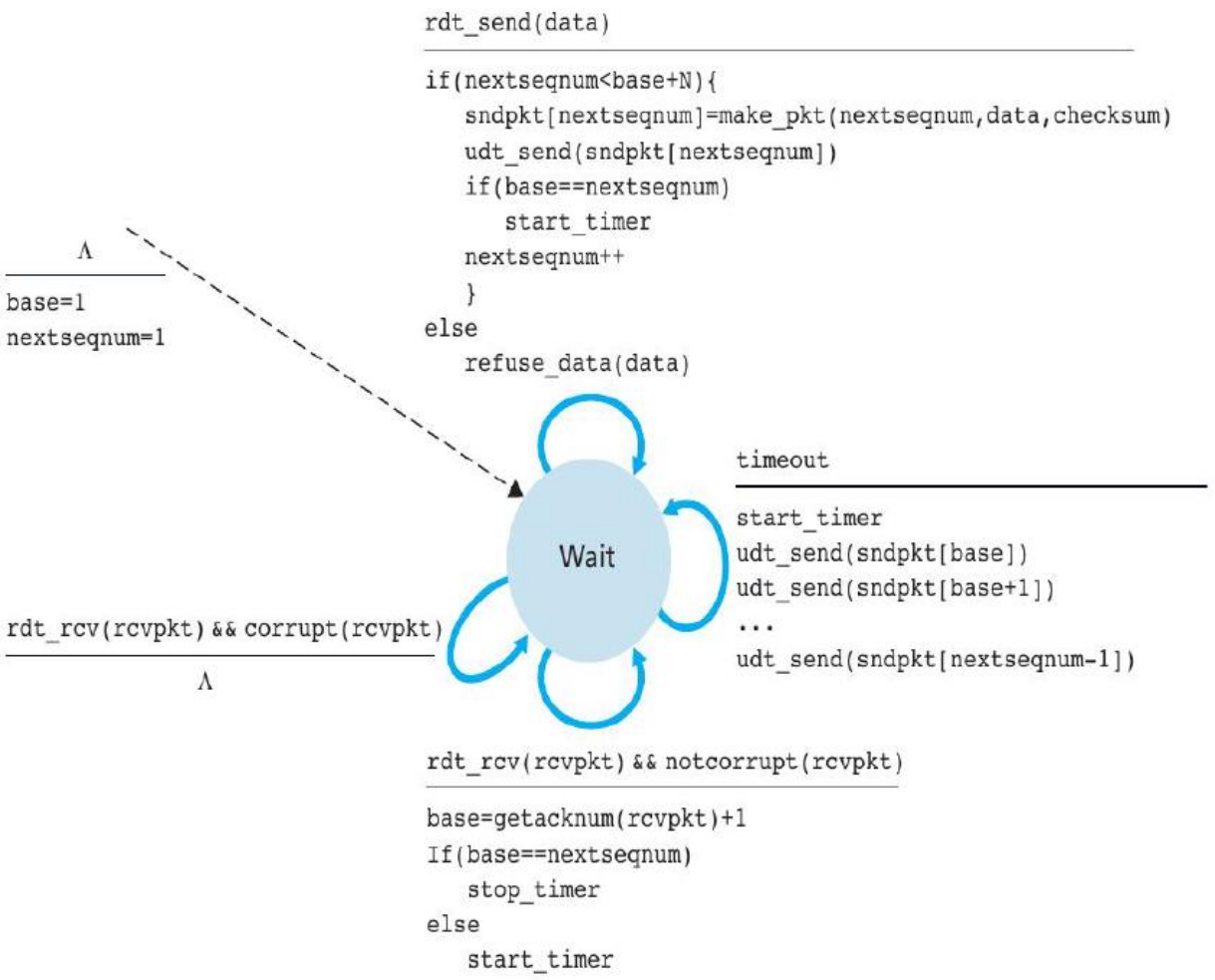


b. Pipelined operation

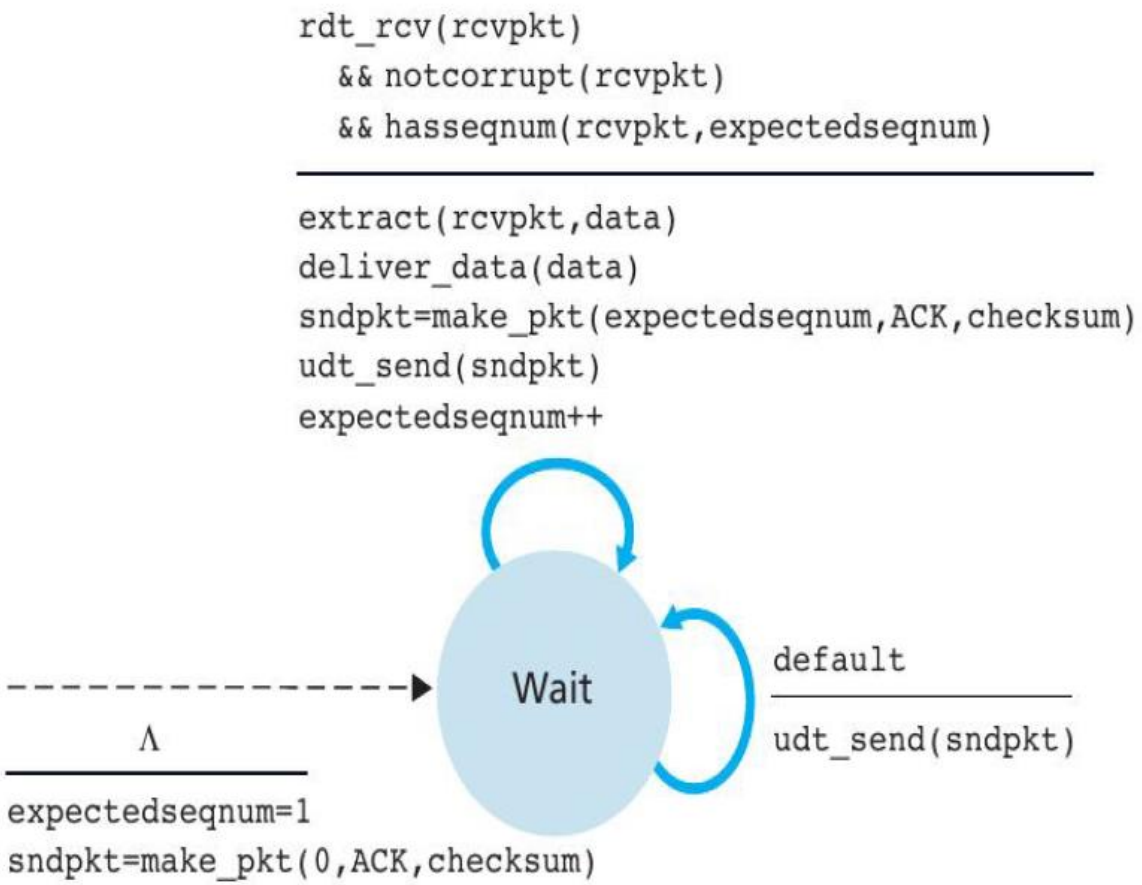
Go-Back-N


base
nextseqnum
Window size
N
Key:
Already ACK'd
Usable, not yet sent
Sent, not yet ACK'd
Not usable

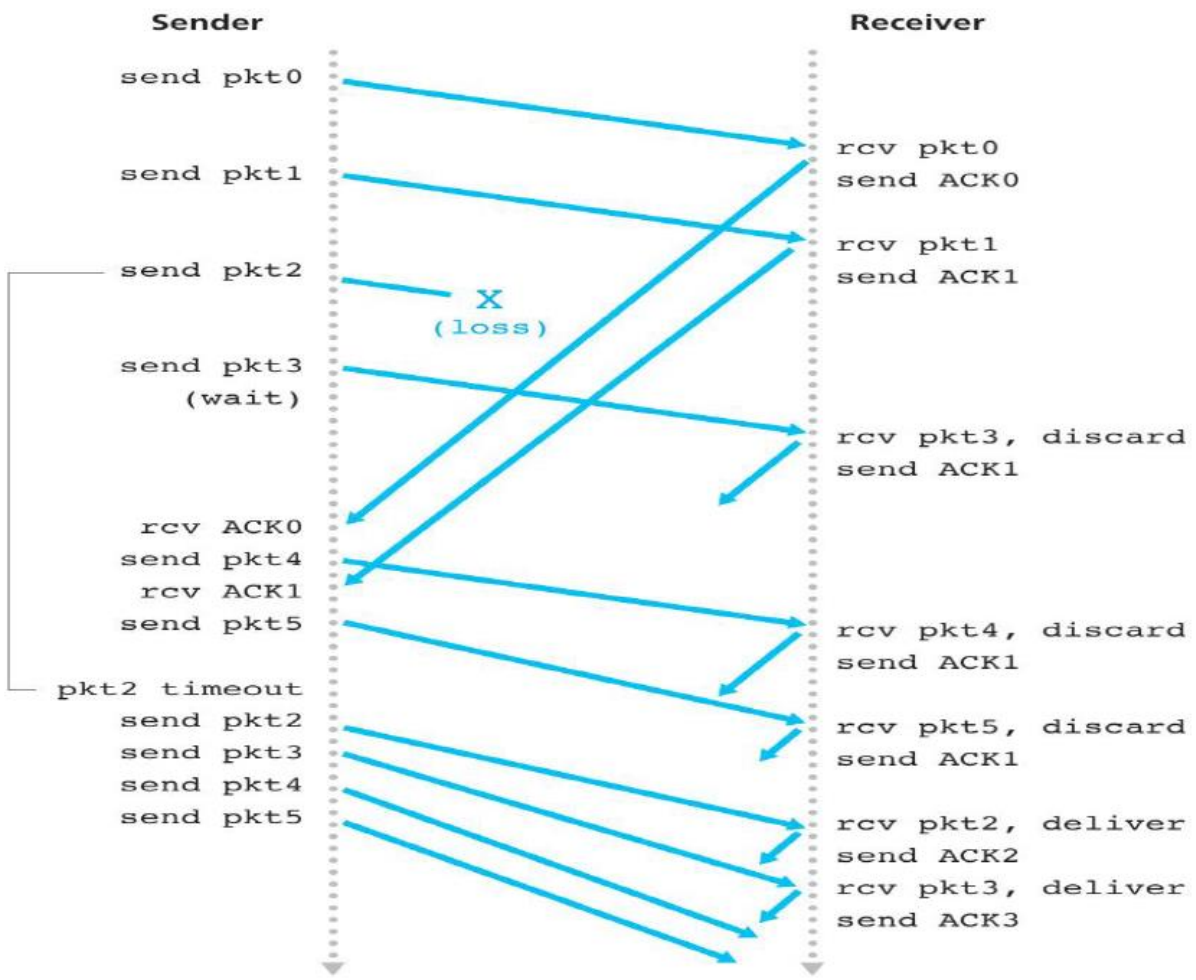
GBN Sender


rdt_send(data)
if(nextseqnum<base+N){
sndpkt[nextseqnum]=make_pkt(nextseqnum,data,checksum)
udt_send(sndpkt[nextseqnum])
if(base==nextseqnum)
start_timer
nextseqnum++
}
else
refuse_data(data)
Λ
base=1
nextseqnum=1
Wait
timeout
start_timer
udt_send(sndpkt[base])
udt_send(sndpkt[base+1])
...
udt_send(sndpkt[nextseqnum-1])
rdt_rcv(rcvpkt) && corrupt(rcvpkt)
Λ
rdt_rcv(rcvpkt) && notcorrupt(rcvpkt)
base=getacknum(rcvpkt)+1
If(base==nextseqnum)
stop_timer
else
start_timer

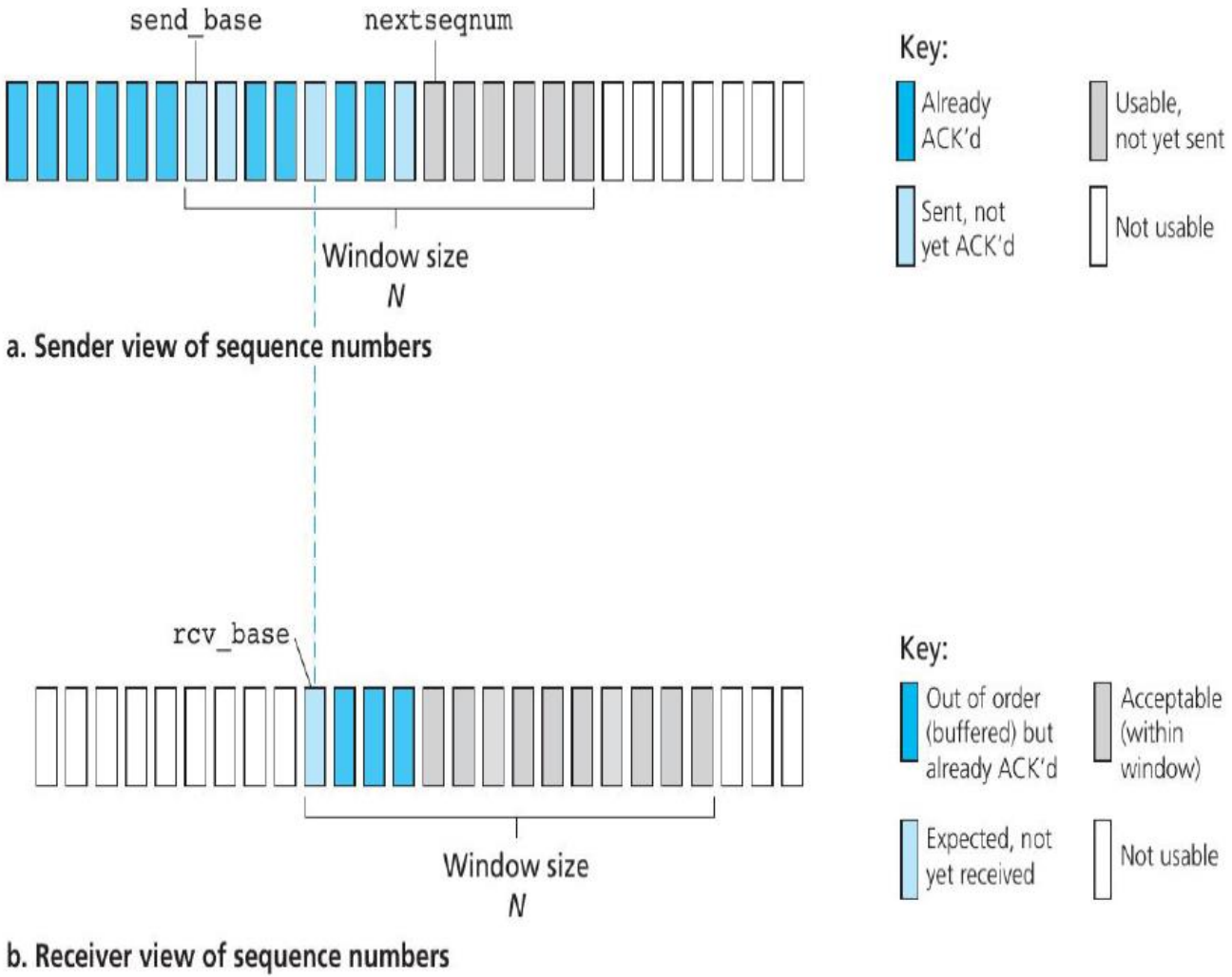
GBN Receiver



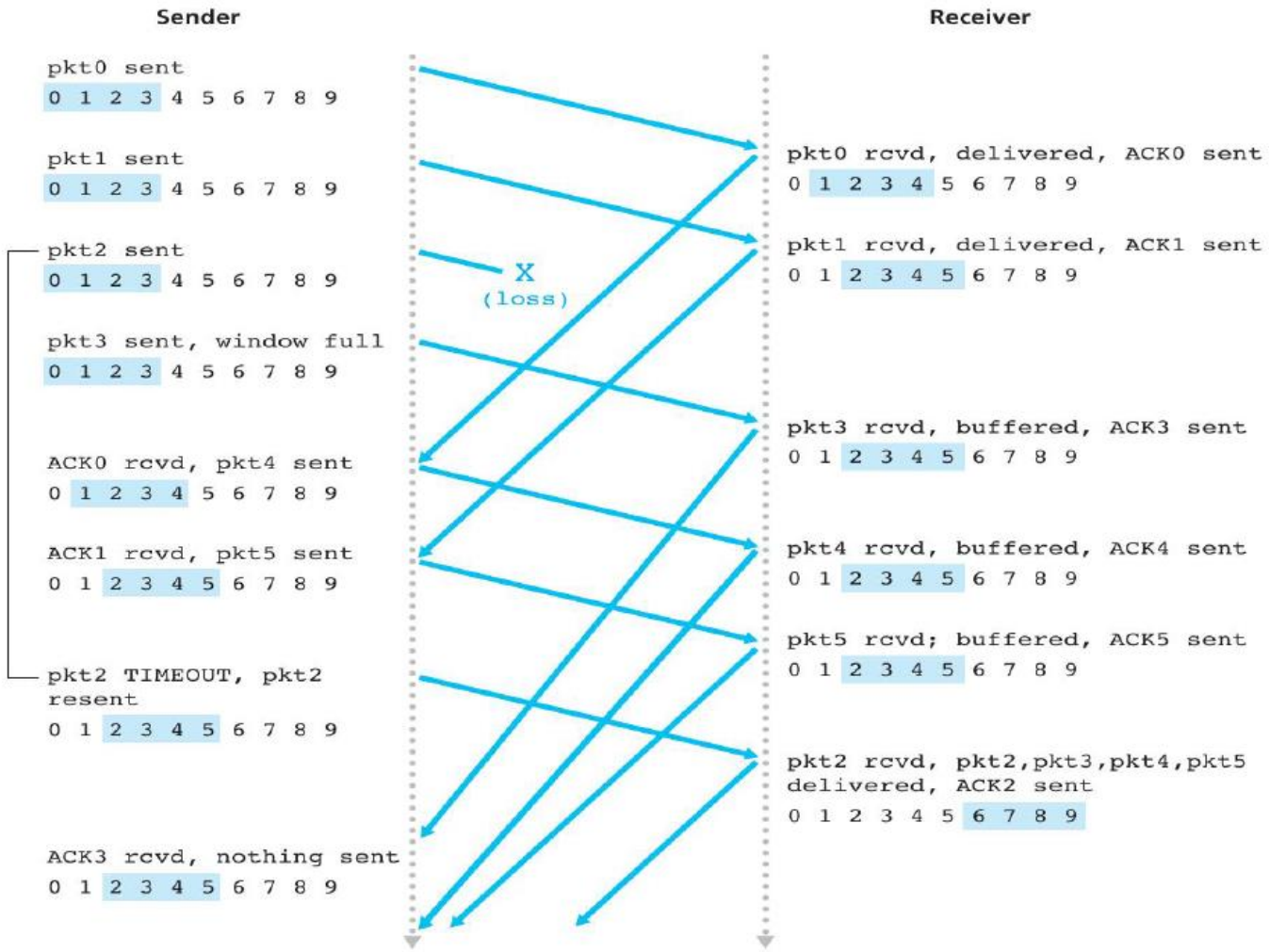
GBN Operation



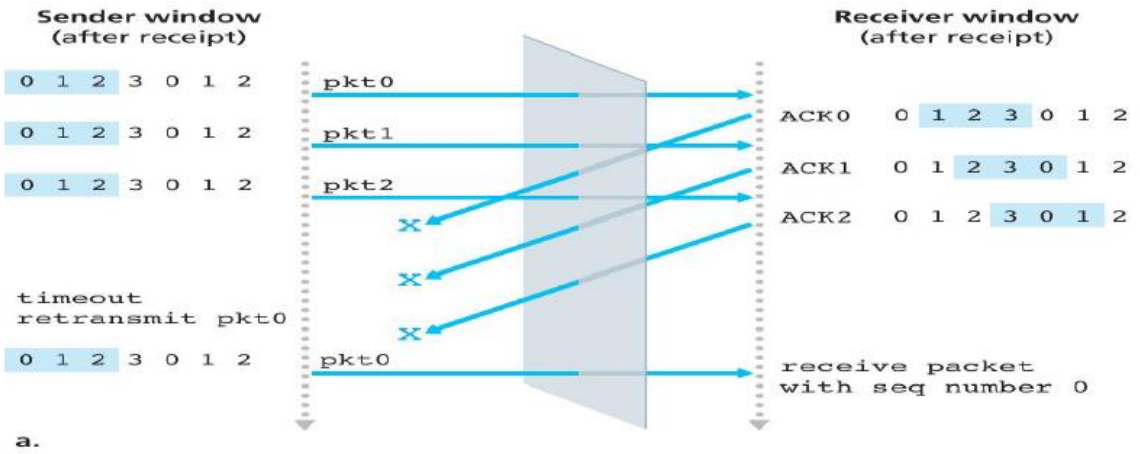
Selective-Repeat

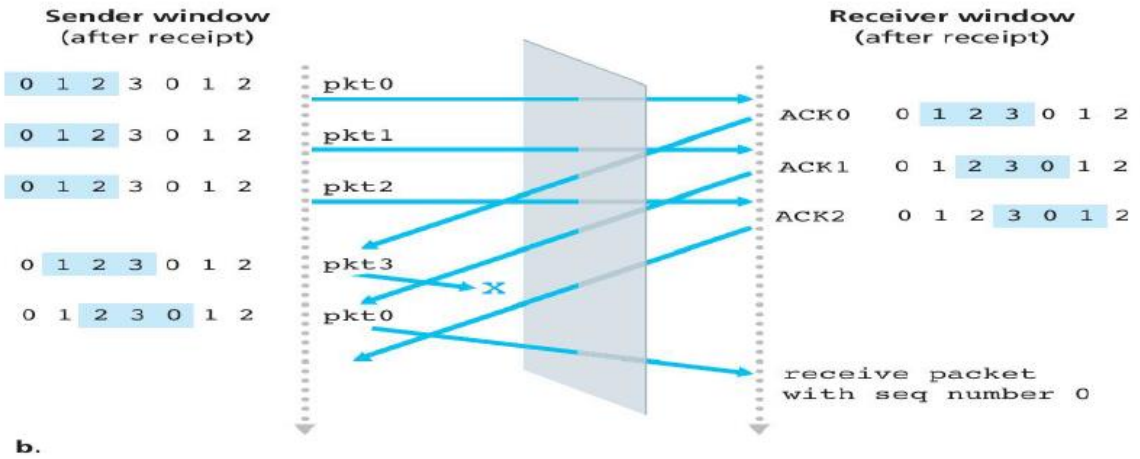


SR Operation



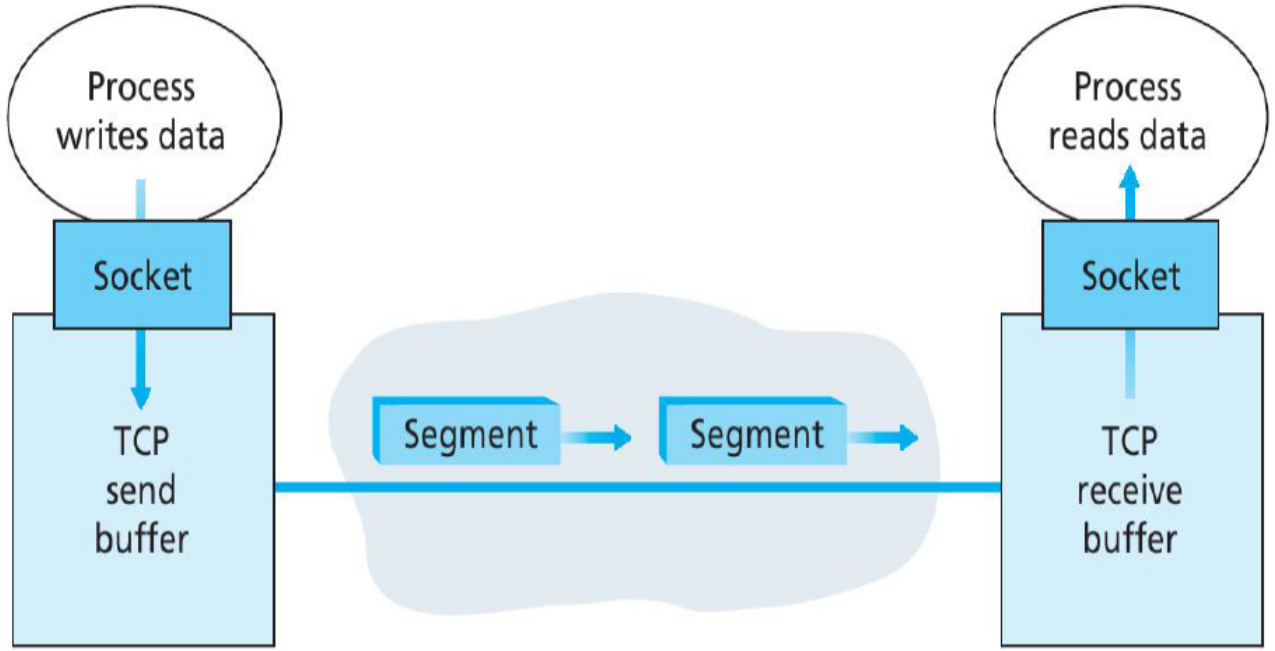
Window Size in SR



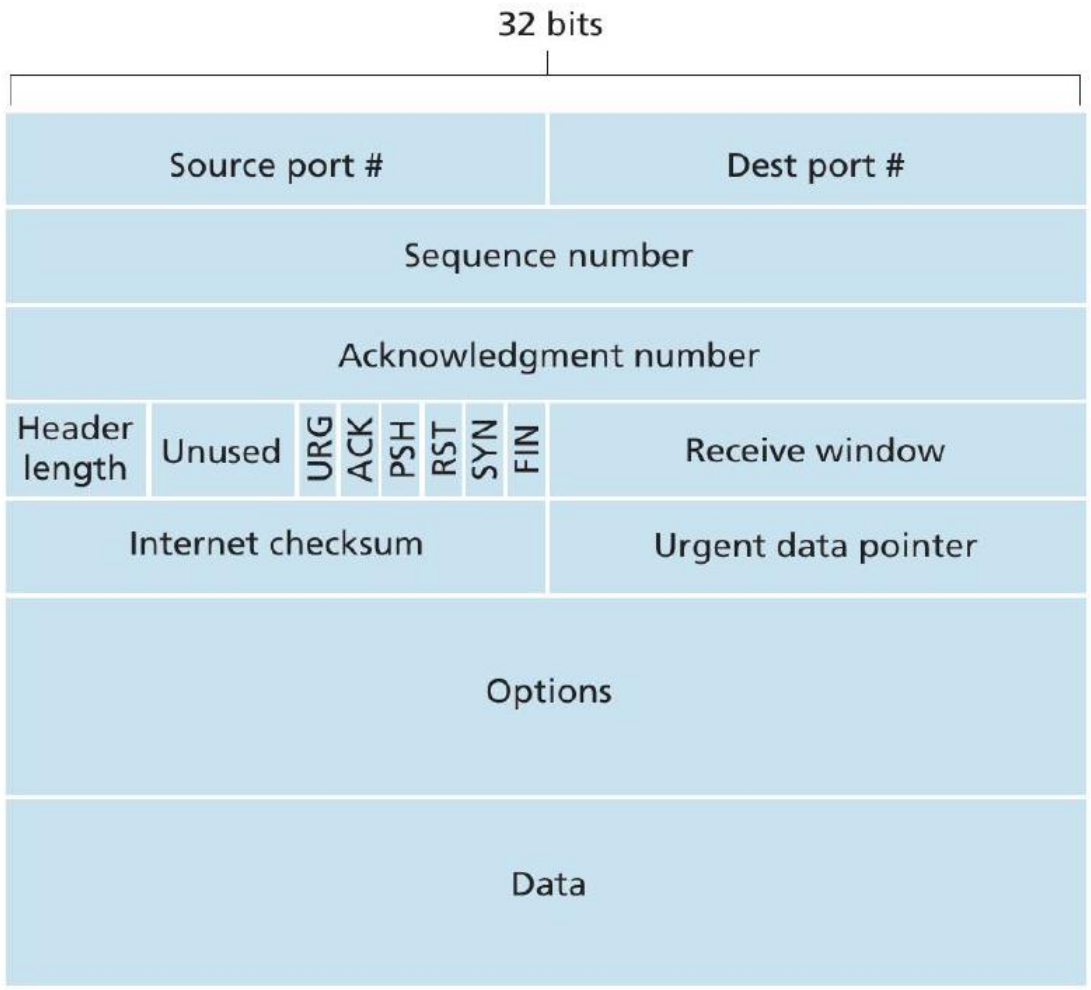


TCP

- TCP is a full duplex service
- No multicasting
- Maximum segment size (MSS) is the maximum amount of data that a TCP segment can contain.



TCP Segment

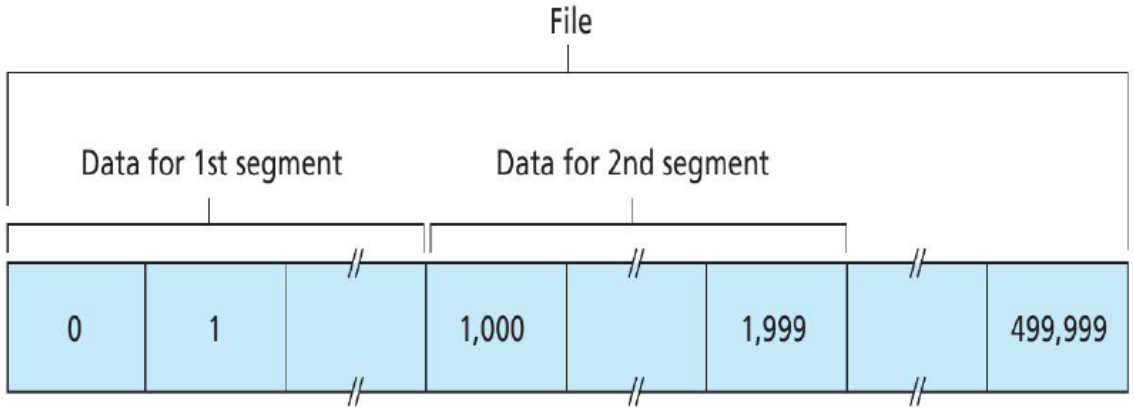

32 bits
Source port #
Dest port #
Sequence number
Acknowledgment number
Header length
Unused
URG
ACK
PSH
RST
SYN
FIN
Receive window
Internet checksum
Urgent data pointer
Options
Data

TCP Segment

- The 16-bit receive window indicates the number of bytes that a receiver is willing to accept
- Header length filed is 4-bytes, specifies the length of the TCP header in 32-bit words.
- Options are used to negotiate MSS, include time-stamping, etc.
- The flag field contains 6 bits, RST, SYN, FIN are used for connection setup and teardown.
- PSH indicates that data has to be sent to upper layers immediately.
- URG is used to mark the segment as urgent, when it is on there will be a 16-bit urgent data pointer filed at the end of urgent data.

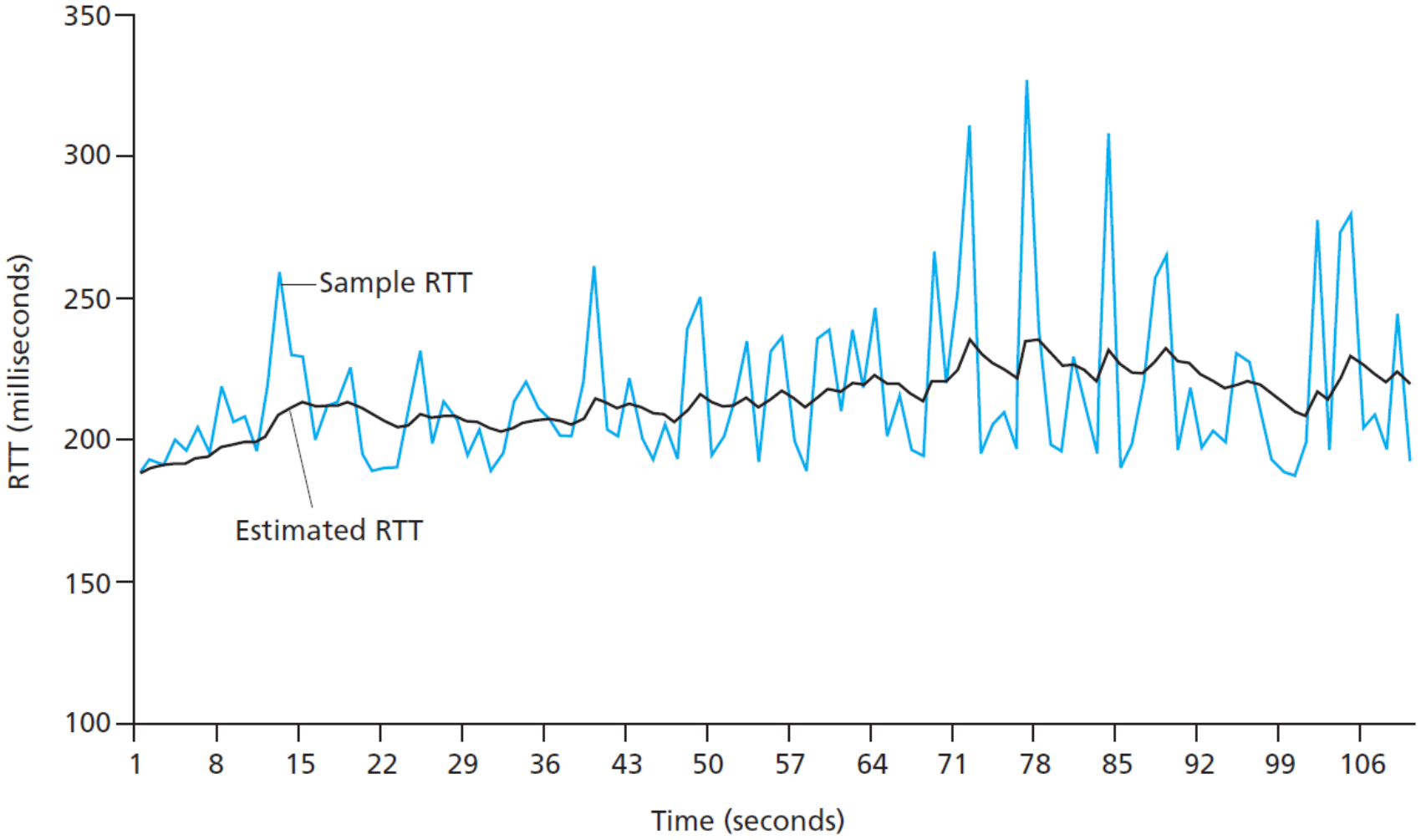
TCP Sequence Numbers

- The sequence number of a segment is the byte-stream number of the first byte of data.
- The acknowledge number is the sequence number of the next byte that is receiver is expecting from source.
- TCP provides cumulative acknowledgments; Out-of-order segements?
- Seqence numbers may not always start from '0'.



TCP Timeout

- SampleRTT: RTT of a freshly transmitted packet. Computed for each RTT.
- Exponentially weighted moving average: EstimatedRTT = $(1-\alpha)$EstmiatedRTT + α SampleRTT
- $\alpha = 0.125$
- DevRTT = $(1-\beta)$ DevRTT + β | SampleRTT - EstimatedRTT|
- $\beta = 0.25$
- Timeout = EstimatedRTT + 4. DevRTT


350
300
250
200
150
100
RTT (milliseconds)
Sample RTT
Estimated RTT
1
8
15
22
29
36
43
50
57
64
71
78
85
92
99
106
Time (seconds)