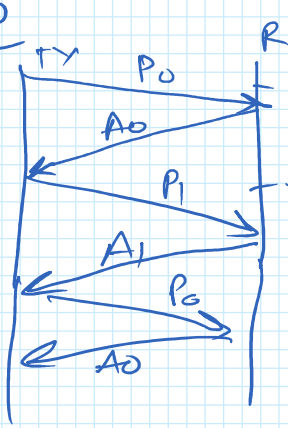
# Lecture 9

07 September 2021 17:04

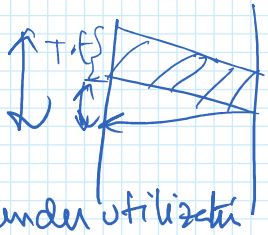
- Recap — stop & Go
- Pipelining based protocol → Go Back N
  → Selective Repeat
- Visual implementation / simulations of ↑ algo in book's website
- Congestion control.

## Stop & Go



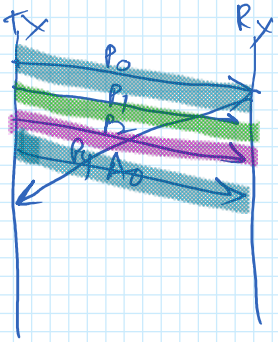

→ < 1% utility
1RTT ⇒ only one pckt is being sent

Improve Stop & Go Perf ? ① → Increase pckt size ≈ RTT




Large pkt size
- solves under utilization
- But increases Packet Error probability

② Pipeline

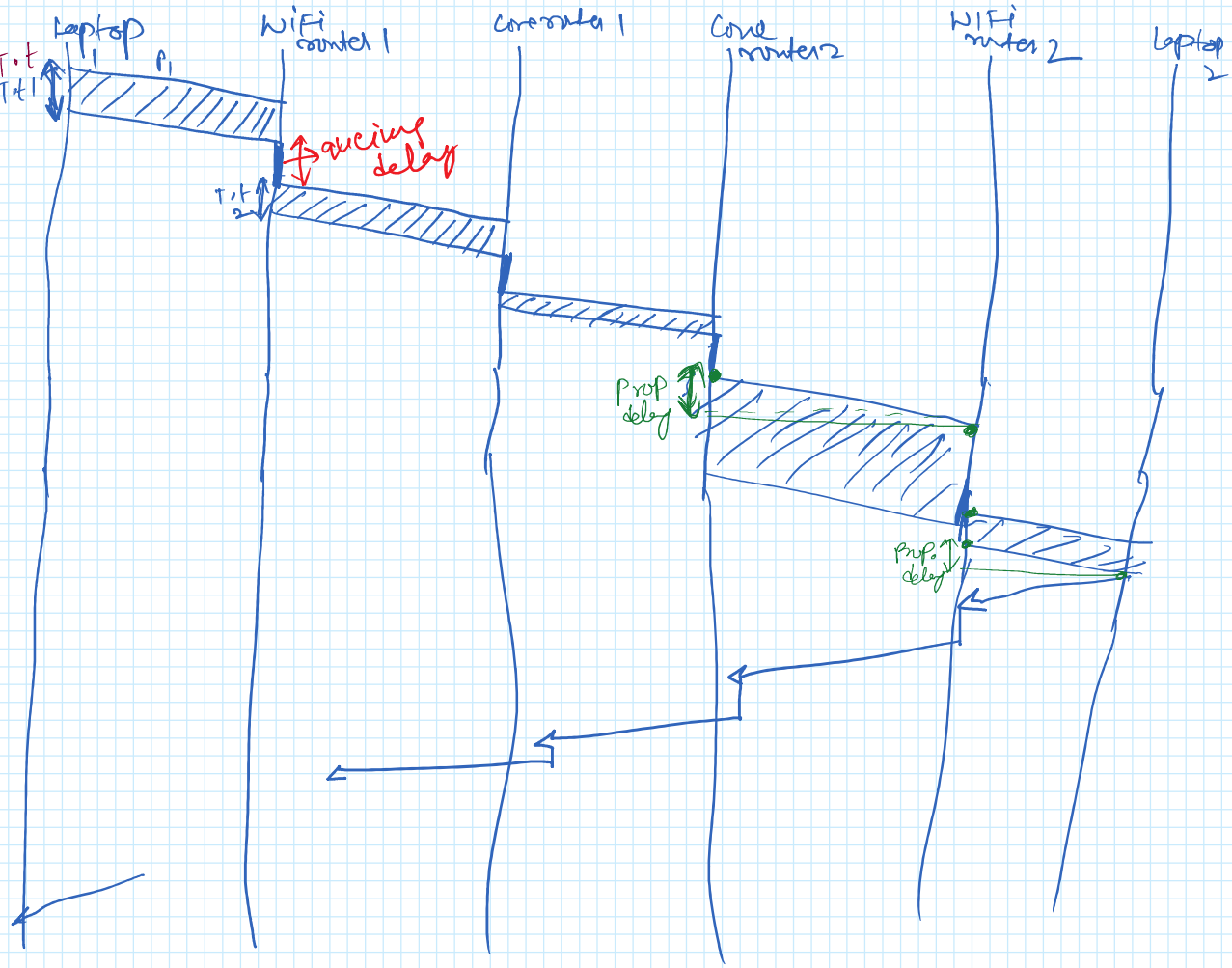



T.t. = transmit time of a pckt

RTT

$$\#\text{ Pckts without ack to send} = \frac{RTT}{T.t.}$$

In Reality timing diagram:

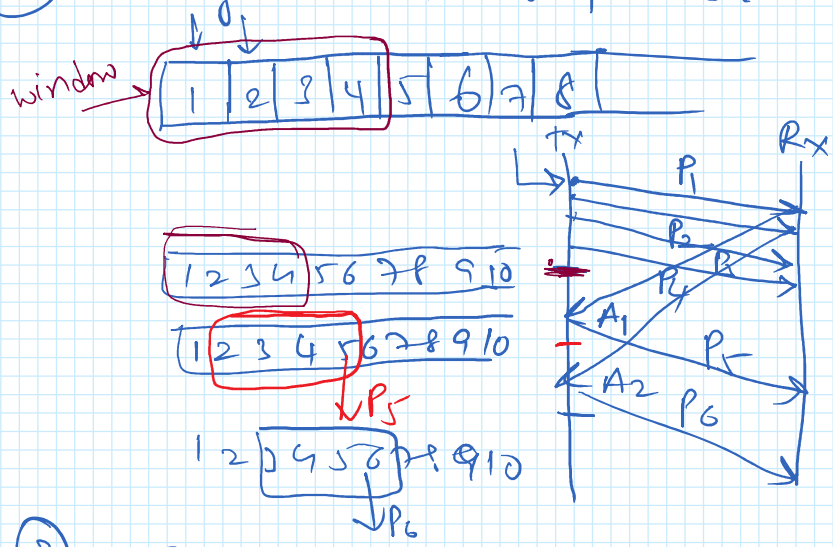



- window size
  - Max
    - No. of unacknowledged pckts that the sending is willing to tolerate
    - In real protocols like TCP, the window size changes (adaptive) based on the current traffic situation

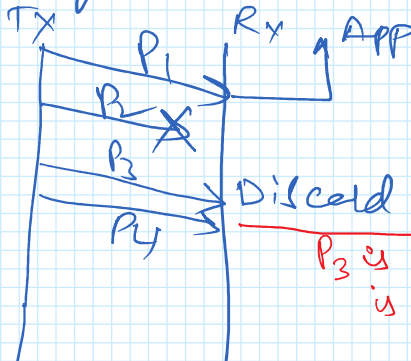
# Go Back N

① – Sliding window based protocol.

window size = 4.



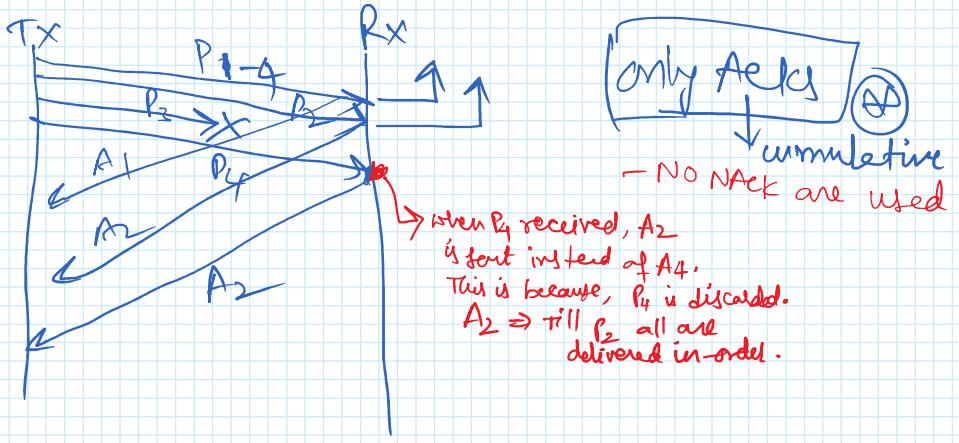

② – Rx. accepts only in-order packets.



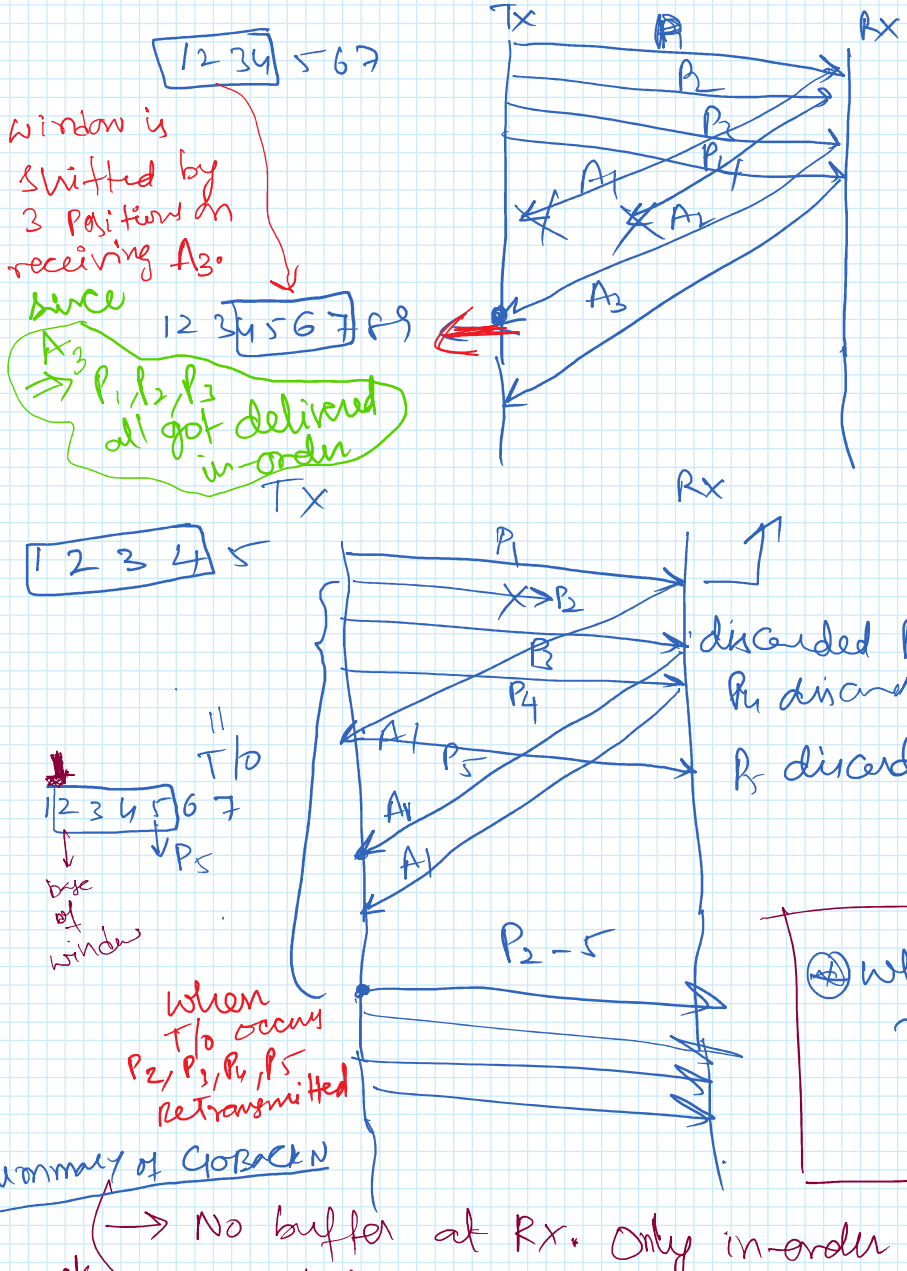

$P_3$ is discarded as the Rx is expecting $P_2$. $P_3$ is received "out-of-order"

③ Cumulative Acknowledgement

$A_4$ ⇒ All packet till $P_4$ are received in-order



## Example :




* When timeout occurs retransmit all pckts in window

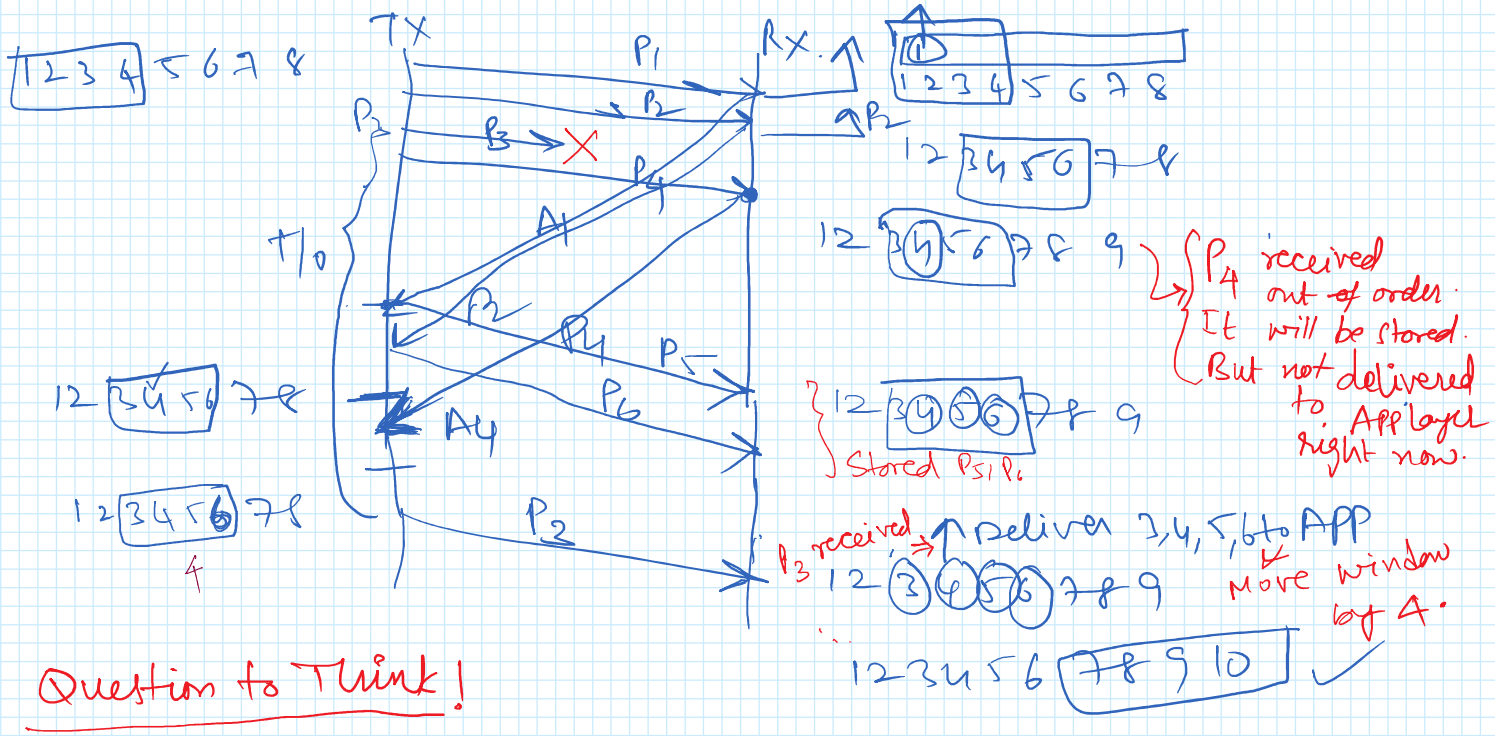
## Summary of GoBackN

GoBack N

- → No buffer at Rx. Only in-order packets will be accepted.
- → TX. maintains a sliding window
- → T/o ⇒ Resend all pckts in window
- → cum. Ack ⇒ all pckts till that are received in-order

# Selective Repeat Algorithm

- Rx accepts out-of-order packets.
- Rx has buffer and a window
  - ↓ determines how many out-of-order pckts it can store.
- Acks are specific to pckt received. (i.e, NOT cumulative Ack)




## Question to Think!

- Should Rx & Tx window size be same?!

Case (i) If Rx window size > Tx window size ⇒ No use since extra space unused because Tx is not going to any new pockets

Case (ii) Tx window size > Rx wsize.

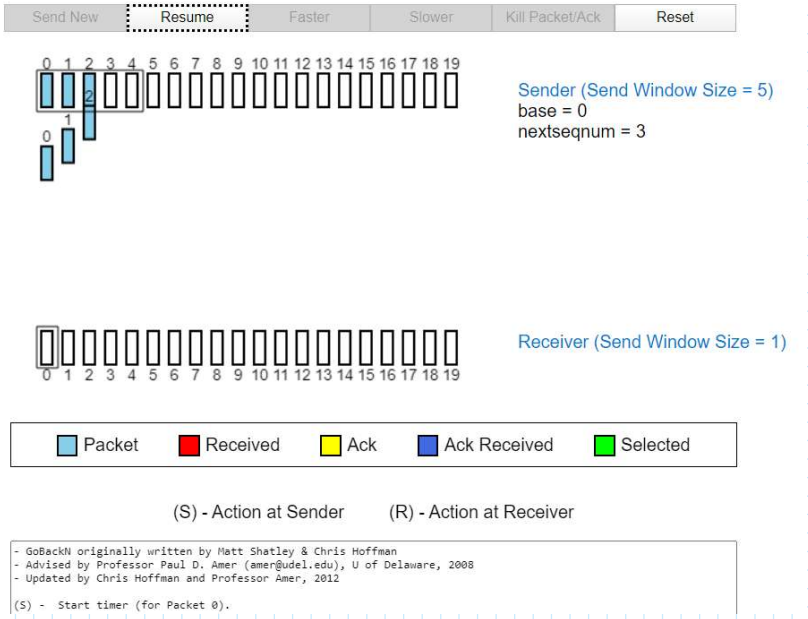
Chapter 3

# Interactive Animations on Book website

URL link ⇒ https://wps.pearsoned.com/ecs_kurose_compnetw_6/216/55463/14198702.cw/index.html

## Go-Back-N Protocol.

This interactive animation brings to life the Go-Back-N protocol. In this demo, the sending window limit
New". This action will begin moving data packets between sender and receiver. To simulate loss, selec
easier. Speed up or slow down the simulation by clicking "Faster" or "Slower". BE PATIENT for retrans




Problem in chapter 3 of Kurose & Ross Book 6 edition (Indian version)

R12

Send five packets
kill first pckt.

GoBack N