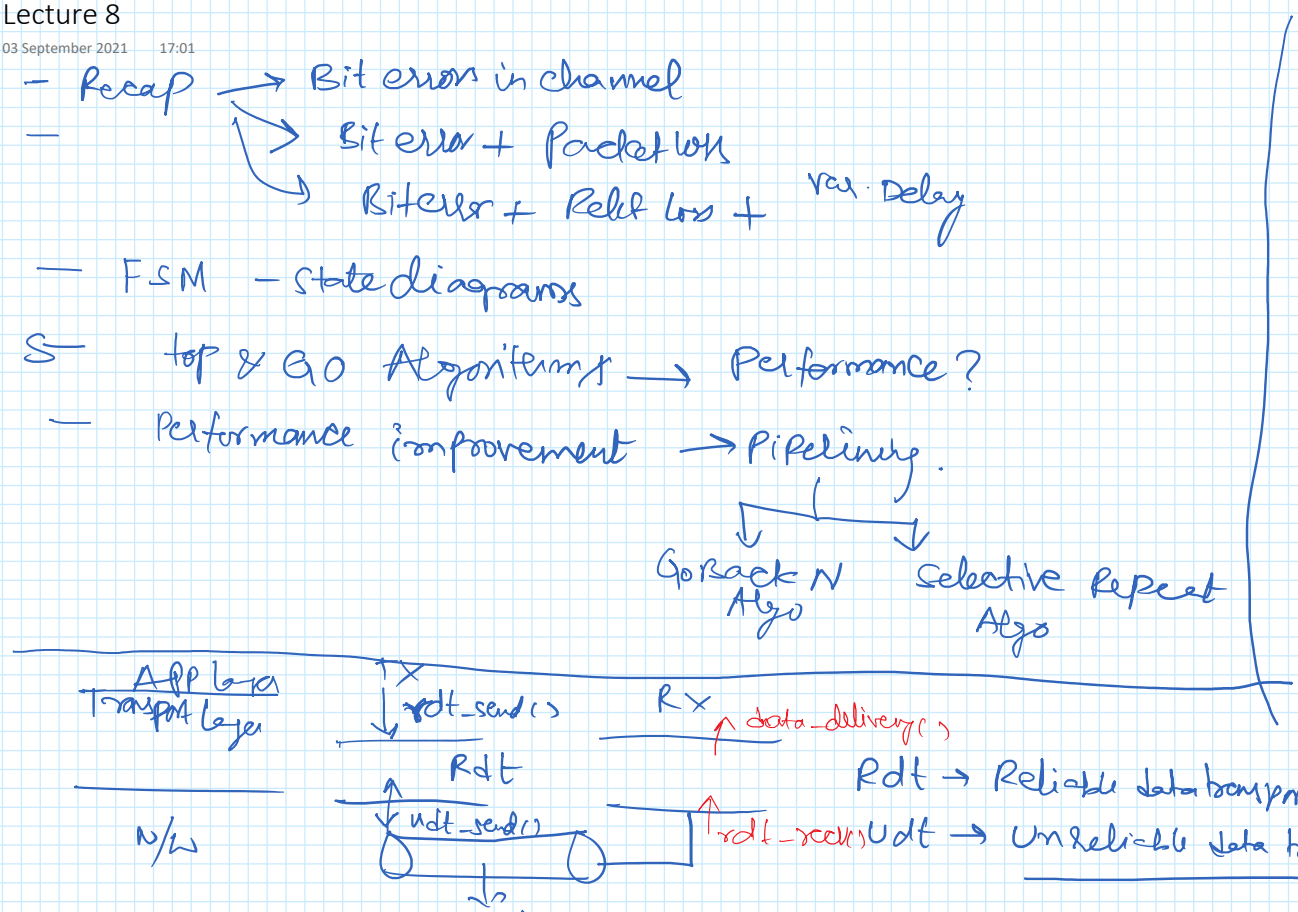
Lecture 8

03 September 2021 17:01

- Recap → Bit error in channel
- → Bit error + Packet loss
- → Bit error + Pckt loss + var. Delay
- FSM – state diagrams
- top & GO Algorithms → Performance?
- Performance improvement → Pipelining.
  - → Go back N Algo
  - → Selective Repeat Algo



Notation in Book

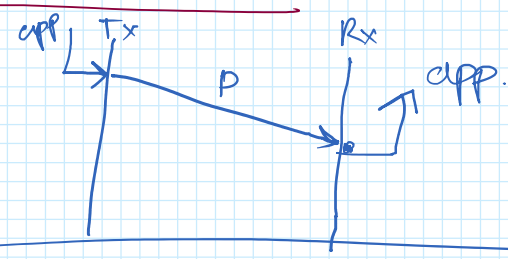
Transp pckt → segment

N/W Pckt → Datagram

App Data: App → Transport

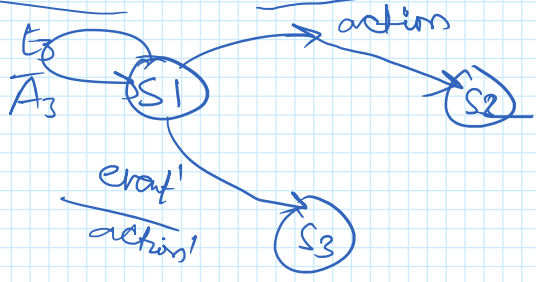
Pckt: Transp → N/W

Reliable channel

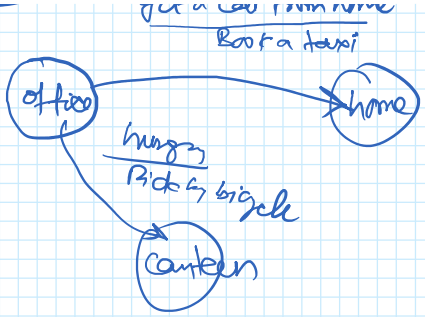


Revisit State Diagrams Concept:

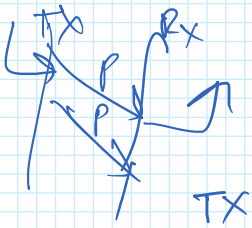
- states
- Transitions
- Event
- Action



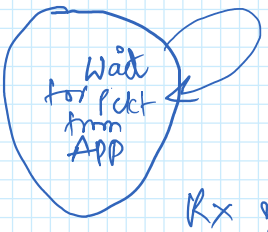
Eg:- get a call from home → Book a taxi



Case I
Reliable channel

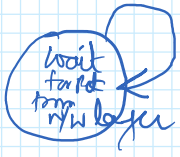


Tx side



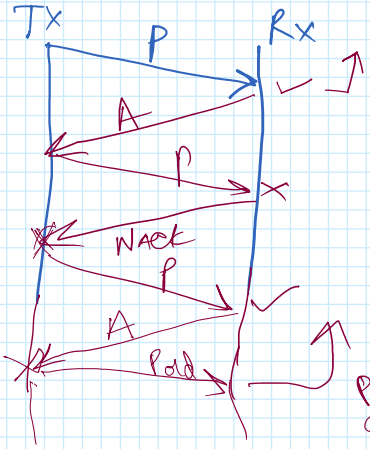
rdt_send(data)
send_pckt = make_pckt(data)
udt_send(send_pckt)

Rx side:



rdt_recv(Pckt)
Extract msg
data_delivery(msg)

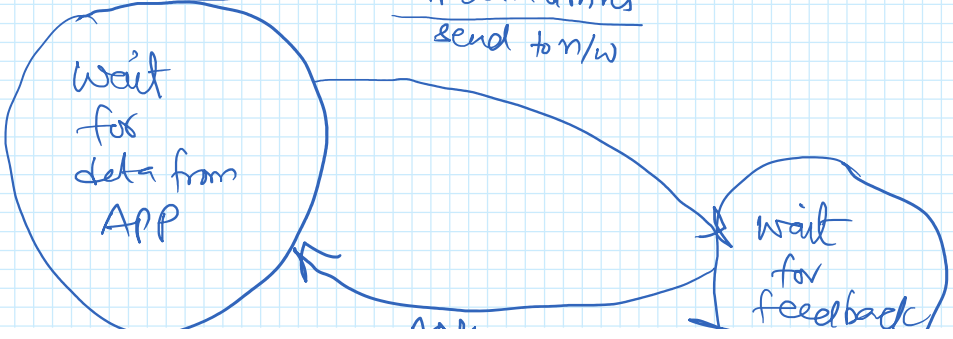
Case II : Bit errors in channel

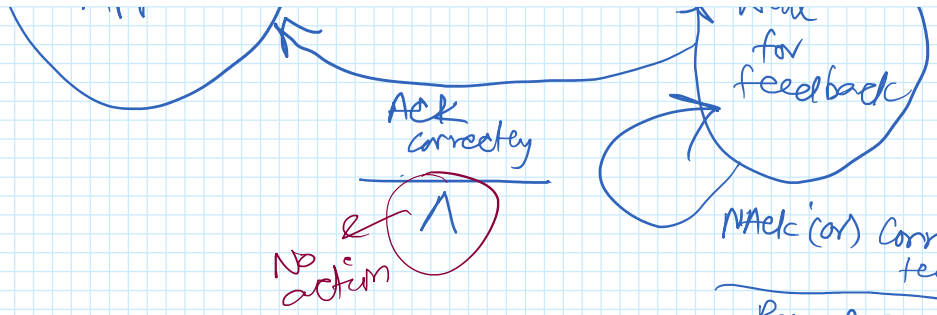


Job of Tx:
- wait for data from app.
- send the pckt to n/w
- wait for feedback
  - ↳ if Ack ⇒ wait for new Pckt
  - if NAck ⇒ Resend Pckt - wait for feedback
  - if corrupt feedback

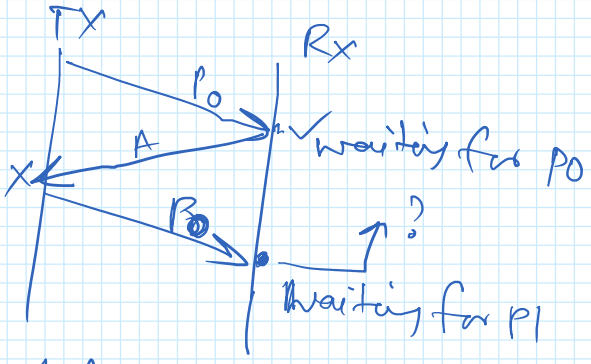
Tx side Diagram:

App data arrives
send to n/w

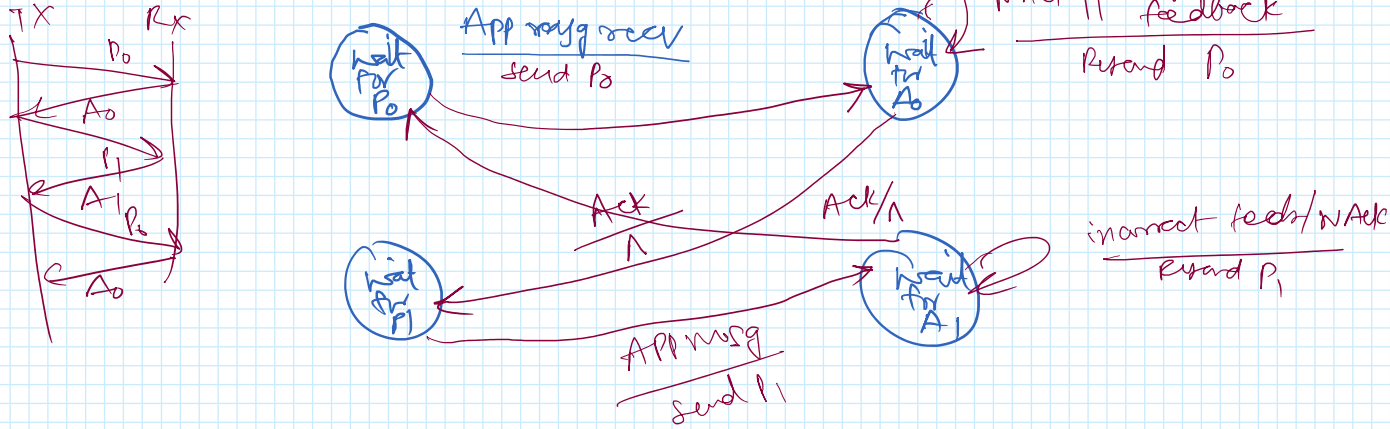




- To solve duplication in Bit error case 1 bit Seq num. P_0 P_1

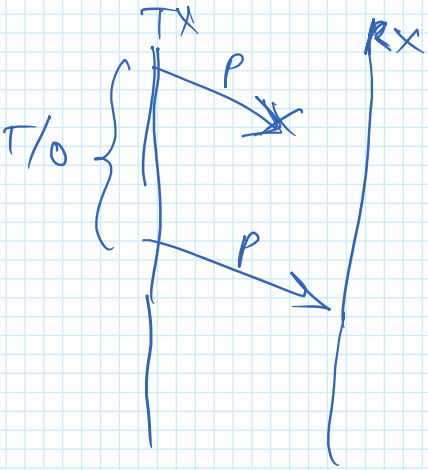


Ideal TX side State Diagram



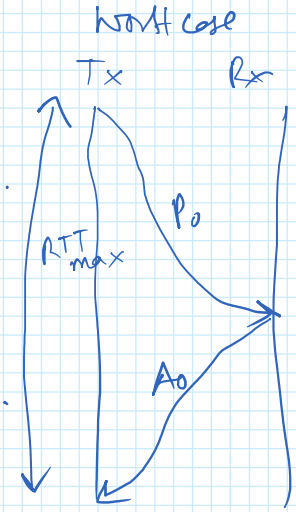
Case III : Bit error + Pckt loss

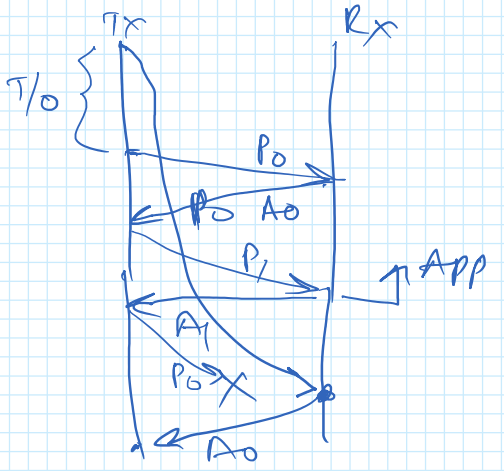
↳ Time out



Case IV : Bit

Case IV : Bits + Loss + Var delay





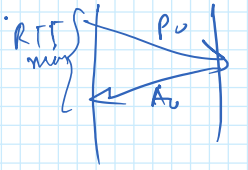
1 bit seq = $P_0\ P_1\ P_0\ P_1 \ldots$

2 bit seq = $P_0\ P_1\ P_2\ P_3\ P_0\ P_1\ P_2\ P_3$

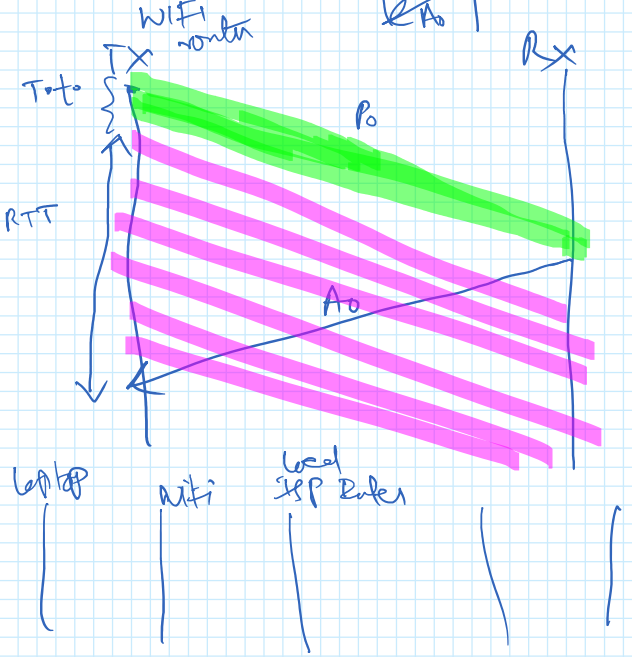
— Increase num. of distinct Pckt sequences.

$$\#seq = \frac{RTT_{max}}{RTT_{min}}$$

solves duplication problem?!



Stop & Go ?



C = 80 Mbps

L = 1 KB

$$U_{sender} = \frac{T.t}{RTT + T.t}$$

$$T.t = \frac{L}{C} = \frac{10^3 \times 8 \text{ bits}}{80 \times 10^6 \text{ b/s}} = 10^{-4}\ s$$

Typical RTT = 0.1 ms ≃ 20 ms

$$U_{sender} = \frac{0.1}{\quad}$$

$U_{header} = \frac{0.1}{20.1}$

$\approx 0.5\%$

$< 1\%$

Performance improvement

→ pipelining

→ Increase pkt size. → any issues?

- $p_b = 1/2$ ✓
- 2bit $P_{pck} = \left(\frac{1}{2}\right)^2$

↓

Low pkt size

- computation/memory
- Header