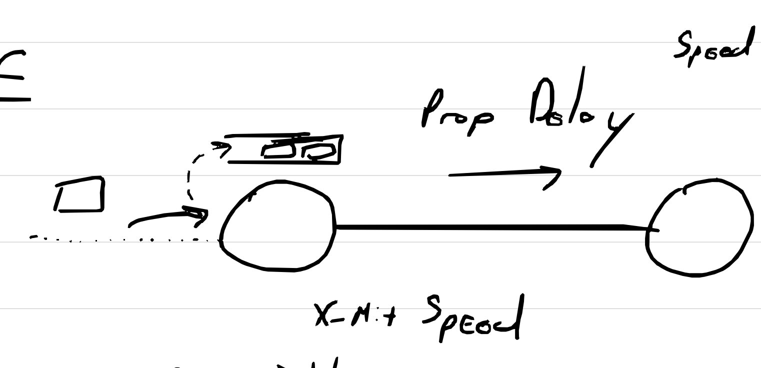
Lecture 2 - 1/14/26

Blog → B/W Links / Properties

G = V, E

Node
Link



Routing Table

Delay ≃ Latency
① Prop Delay — MS
② Transmission Delay — Bits/second
③ Processing Delay — TCAM / Table
④ Queuing Delay

RTT — Ping

What dominates?

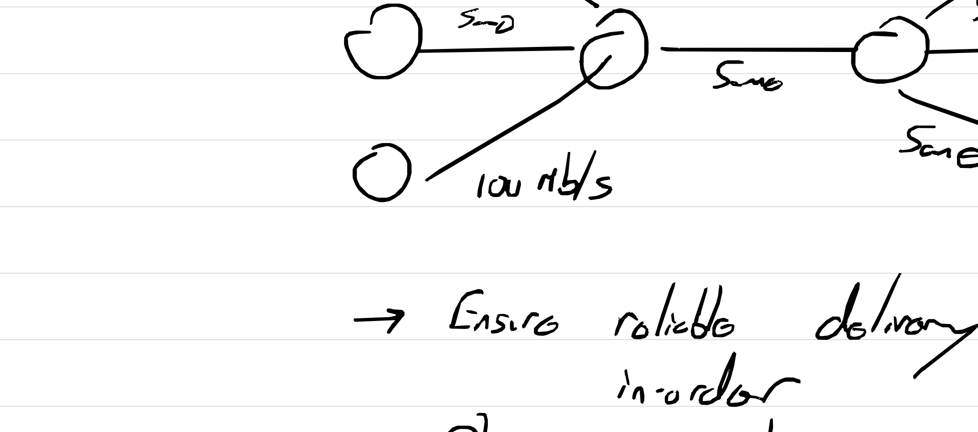
Jitter

Bandwidth ≃ Link Speed — bps

Loss — P_L (↑ Loss) — $1 - P_L$ (Success)

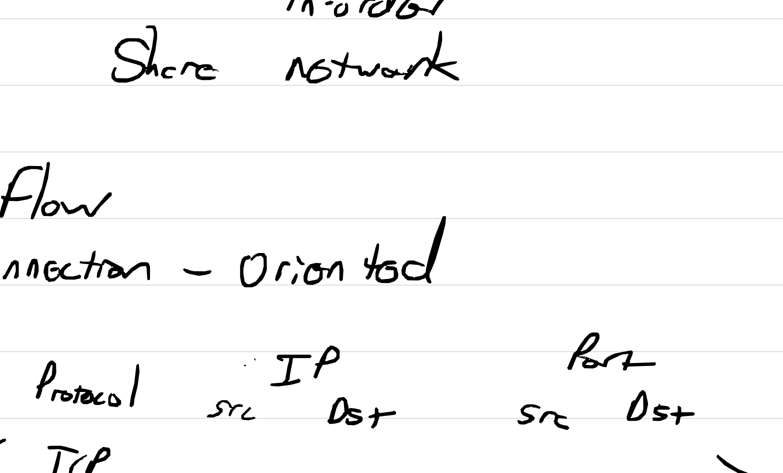
Sources: Queue → Overflow
Transmission → Usually wireless — Next Week

1 Link — Bi-Directional
Symmetric



Multiple Links

Network



→ Ensure reliable delivery
in-order
Share network

TCP — Flow
Connection - Oriented

5 Tuple < Protocol (TCP / UDP), IP (Src, Dst), Port (Src, Dst) >

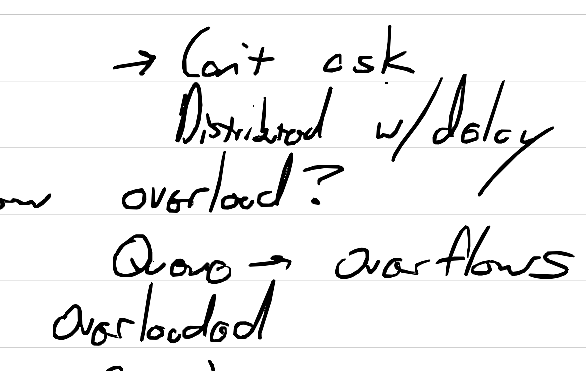
Dotted Notation (IP) — 16 bit, 0-65535 (Port)

Client → Server
connects

IP — Node — Street
Port — Application / Service — Apt #

80 — HTTP
443 — HTTPS
< 1k → Root

Reliable
In-order
"Fair"



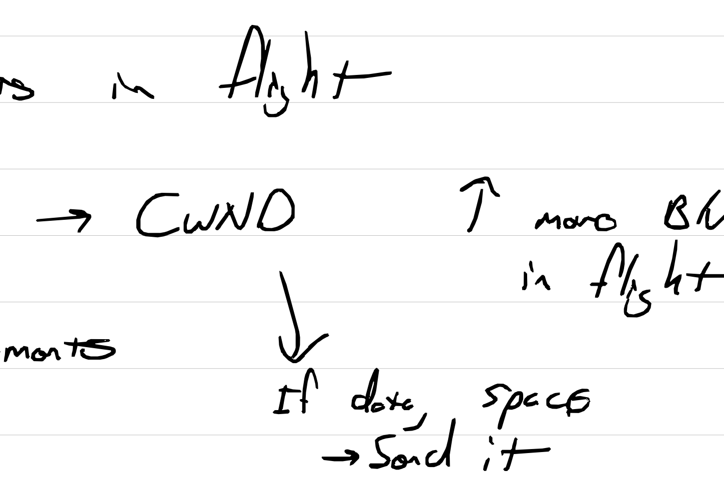
→ GET
← Here you go

Network is dumb → Can't ask
Distributed w/ delay

How do we know overload?
Congestion — Queue → overflows
∴ Loss → Overloaded Signal

TCP — Lose pkts
Timer

Server → Client



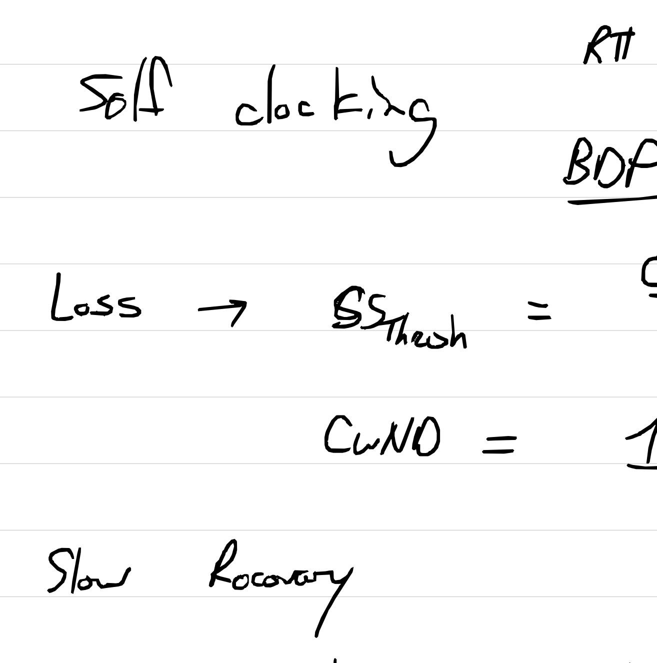
RTO — Multiple of RTT → know that?
Ehhh

Fast → Pkts in flight

How much → CWND — ↑ more BW in flight
↓ If data, space → Send it

TCP → Segments
MSS

Congestion Control → # Pkts in Flight
CWND



AIMD → Test slowly
React quickly

Slow Start ← start w/ growth
CA

RTT ↓ Quicker

Self clocking

BDP

Loss → $SS_{Thresh} = \frac{CWND}{2}$

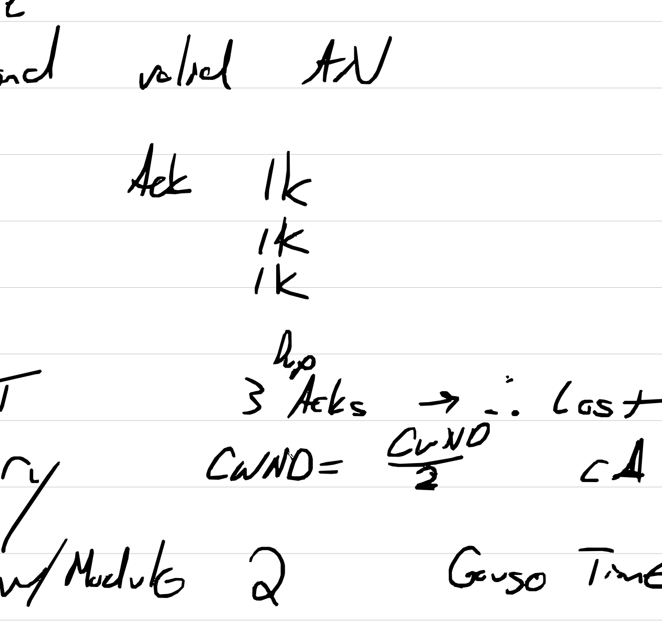
$CWND = 1$

Slow Recovery

Could we know earlier?

Cumulative

In-Order
SN
AN



Send SN
Ack also

Next expected SN

AN → Cumulative
If gap, send valid AN

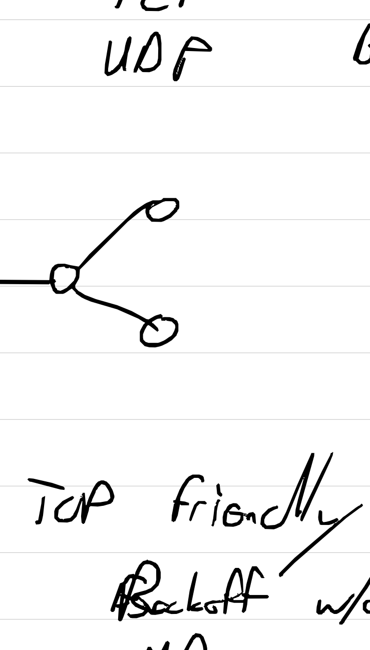
1k →✕ — Ack 1k
2k → — 1k
3k → — 1k

TCP Fast R-T — 3 Dup Acks → ∴ Lost
Fast Recovery — $CWND = \frac{CWND}{2}$ — CA

Variations w/ Module 2 — Guess Time

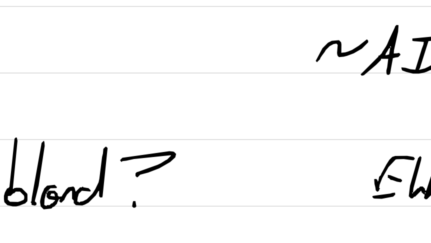
TCP Connection

3-way handshake



Socket → STREAM — TCP — Pipe
DGRAM — UDP — Datagram

Fairness



Jain's Fairness
Max-Min

TCP friendly
Backoff w/ congestion
MD

~AI

Will it blend? — Ehhh

RTT
More aggressive
UDP + TCP

Loss → Wireless
RAGE

Socket Programming

Client ____

Server ____

F/W — Fork / Thread

UDP — send to / recv from — HOLB

Wireshark — tcpdump

Demo