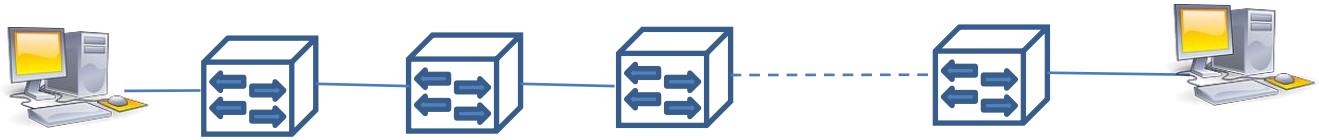
TCP Congestion Control Mechanisms

Kameswari Chebrolu

Seminal Paper: Congestion Avoidance and Control
by Van Jacobson and Michael J. Karels

Congestion Control: Challenge



- Need to estimate W (of sliding window) such that each flow gets its fair share
  - Estimate small → underutilization; Estimate large → Congestion
- W will vary over time
- Congestion Control: Preventing sources from sending too much data too fast and thereby ‘congest’ the network

Idea

- View network as a pipe
- Estimate Bandwidth-delay product (capacity) dynamically
  - Uses the variable Congestion Window (CW) to track it
- Use self clocking to pump packets into the network

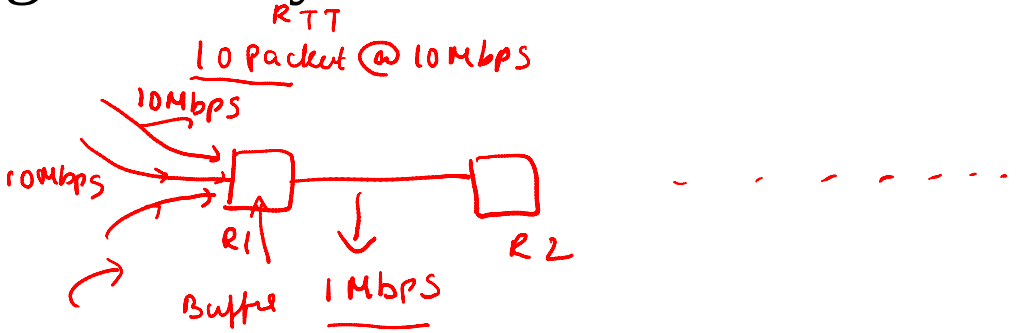
Approach

- Getting to Equilibrium → filling up the pipe
- Conservation at equilibrium
- Adapting to Path

Getting to Equilibrium

BDP – 1, 10, 100, 200

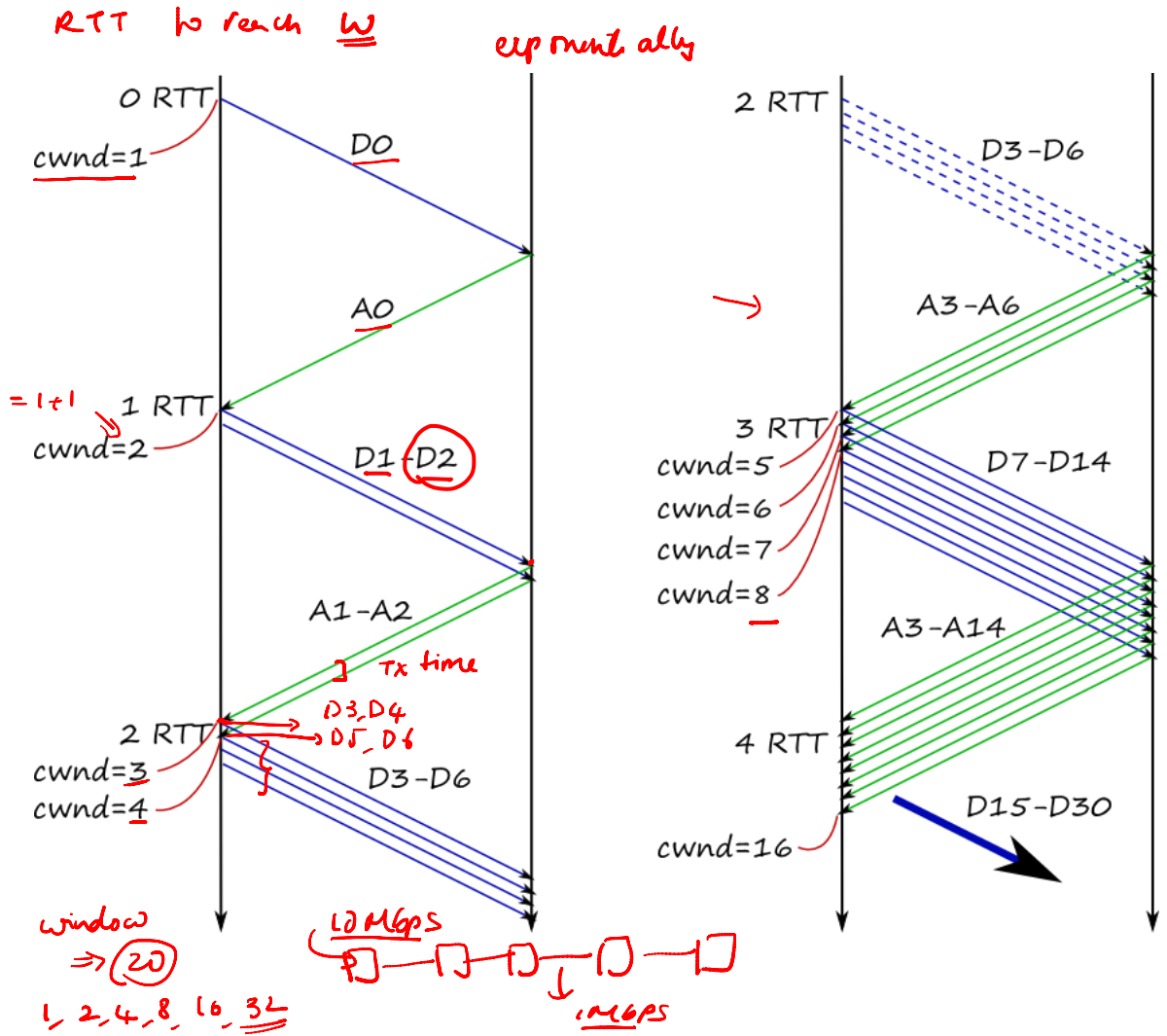
- What value of CW to choose initially?
  - Too large: pushes network into congestion
  - Just right: bursty transmissions can lead to losses



Slow Start

- Add a variable cwnd (congestion window)
  - Captures the number of outstanding data in the network
- At start, set cwnd=1
- On each ack for new data, increase cwnd by 1

- Takes $\log_2 W$ round trip times to get to W
- Sends data at most twice the bottleneck link on the path
- Bound to overestimate capacity at some time in future



Conservation at Equilibrium

- Don't put a packet unless a packet is removed
  - Particularly important when the network is congested
  - Can potentially happen on timeouts → proper RTT estimation crucial
  - Delayed packets should not be interpreted as lost