CSE561 – Congestion Control

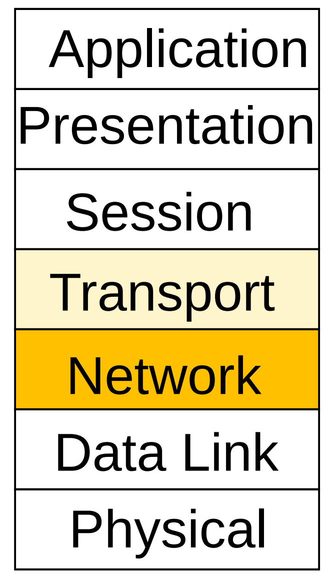
David Wetherall

djw@cs.washington.edu

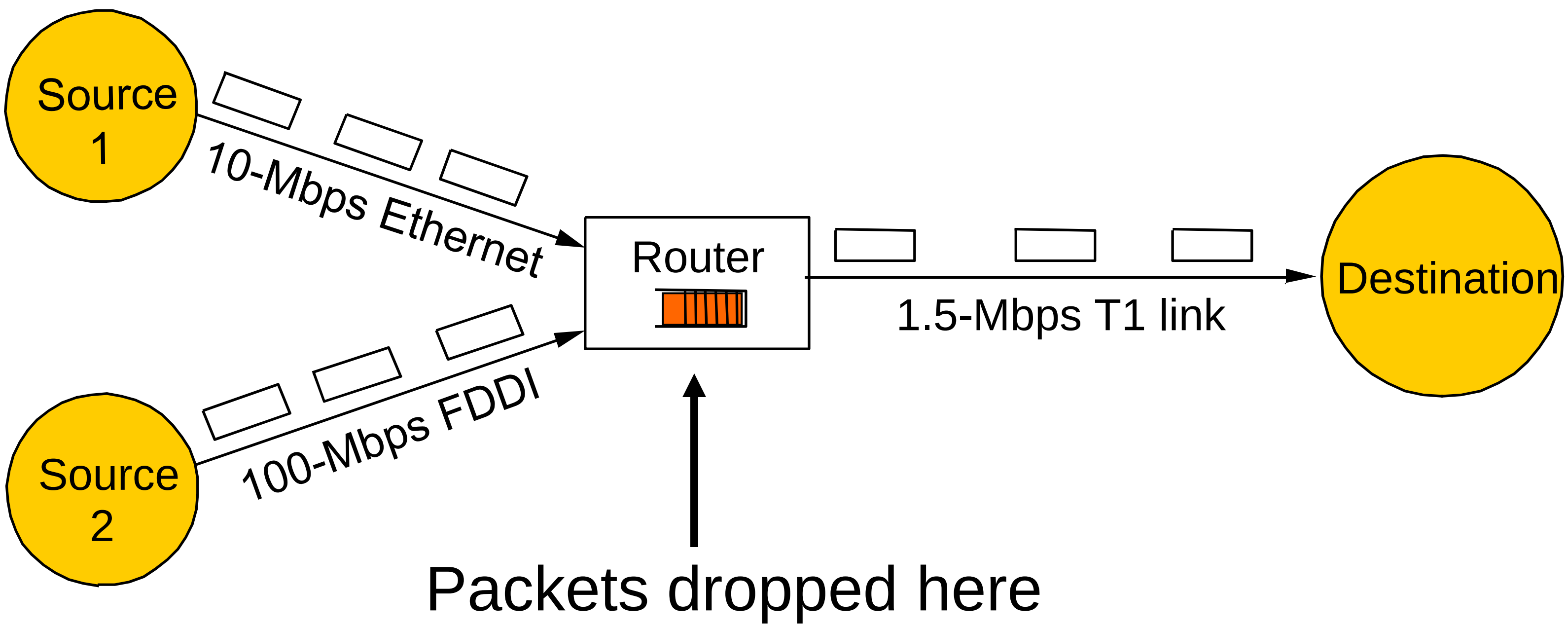
Congestion Control

- Focus:
  - How to share bandwidth between senders

- Congestion
- Fairness
- Bandwidth allocation
- RED/ECN



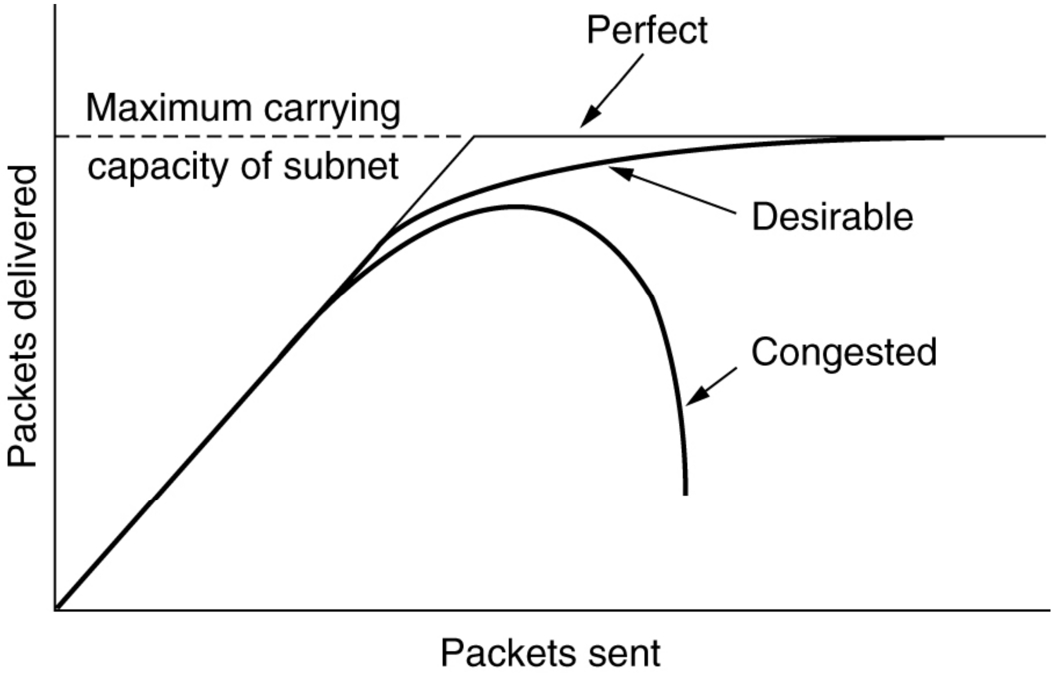
Congestion



- Buffer intended to absorb bursts when input rate > output
- But if sending rate is persistently > drain rate, queue builds
- Dropped packets represent wasted work; goodput < throughput

Effects of Congestion

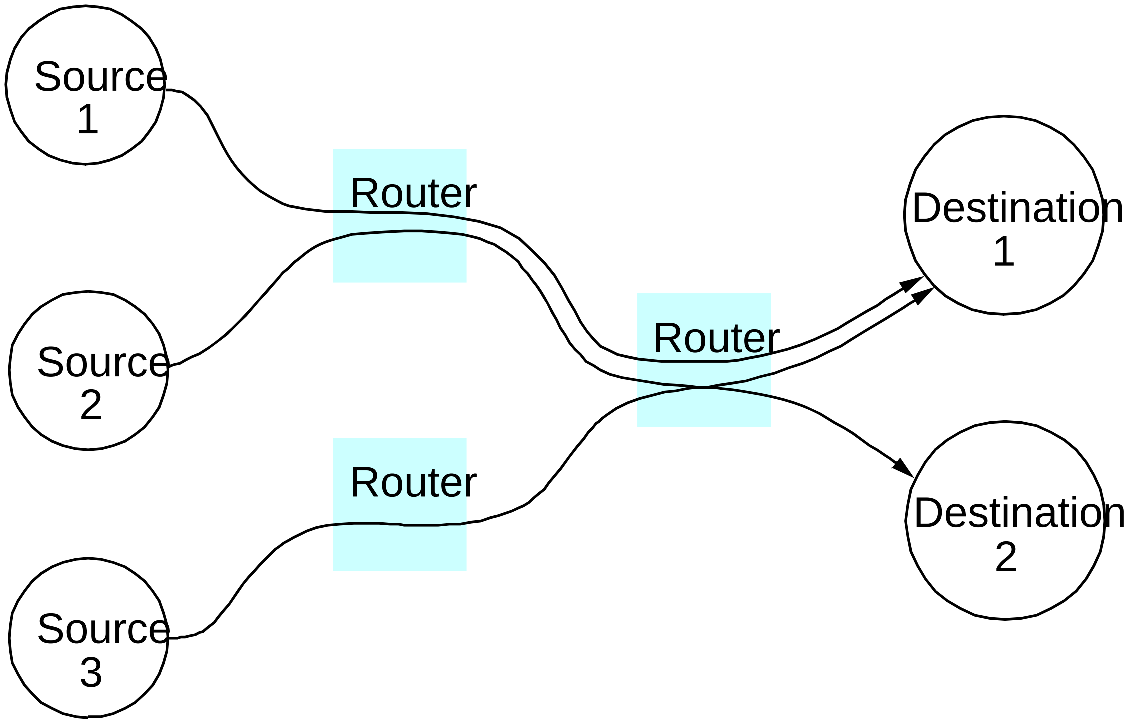
- Draw throughput and delay versus offered load
  - Combination leads to notion of a good operating point
- Also draw goodput versus offered load
  - Notion of congestion collapse



Bandwidth Allocation

- Find a suitable allocation of bandwidth given:
  - a network, workload of traffic flows, and the routes (usually)
- Allocation depends on:
  - A notion of fairness [cf. next slide]
  - A notion of efficiency [cf. last slide]
- Allocation implemented as:
  - Distributed feedback control loop
  - Ouch!

Fairness (min-max fairness)



- Each <u>flow</u> from a source to a destination should get an equal share of their <u>bottleneck</u> link … depends on paths and other traffic

Control Loop

- Traffic is bursty
- Congestion is experienced at routers (Network layer)
- Traffic is controlled at sources (Transport/Network layer)

- The two need to talk to each other!
  - Sources sending more slowly is the only relief
  - Sources sending more quickly is the only way to use the capacity

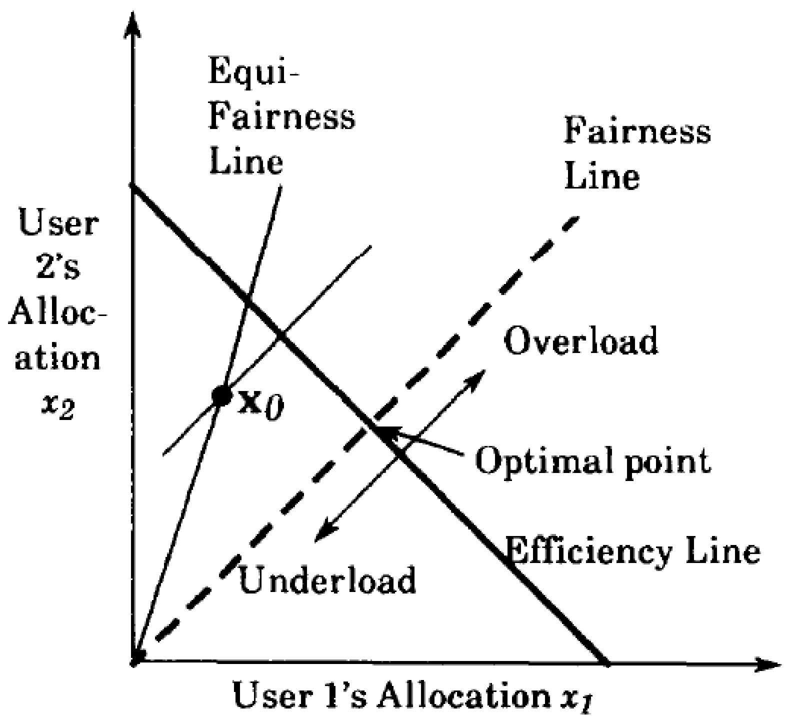
Control Loop Designs

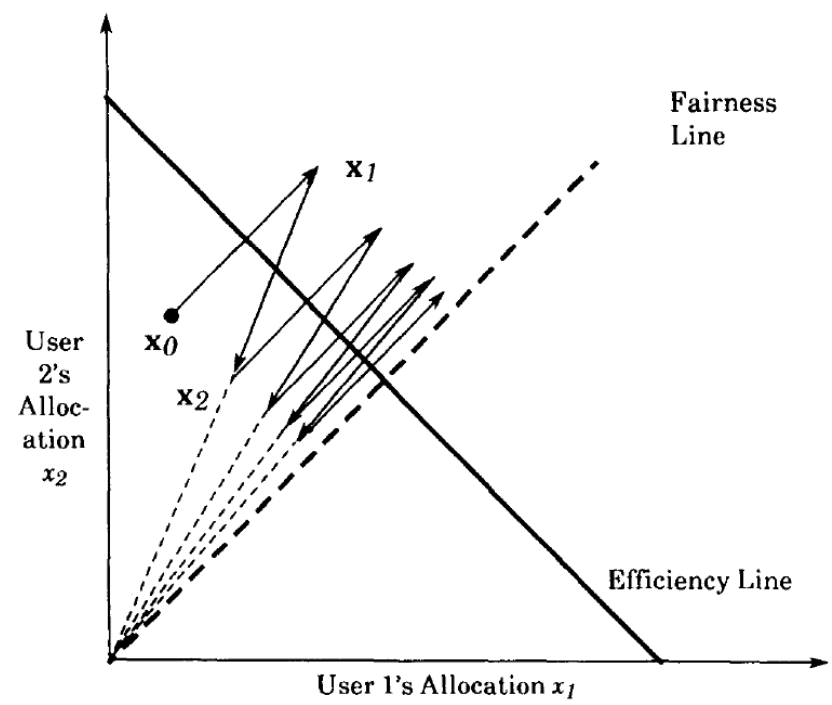
- Open versus Closed loop
  - Open: reserve allowed traffic with network; avoid congestion
  - Closed: use network feedback to adjust sending rate
- Host-based versus Network support
  - Who is responsible for adjusting/enforcing allocations?
- Window versus Rate based
  - How is allocation expressed? Window and rate are related.

- Internet depends on TCP for bandwidth allocation
  - TCP is a host-driven, window-based, closed loop mechanism

AIMD Rationale (Chiu & Jain, 1989)

- AIMD (right) finds the optimal point (left)





Control Loop Feedback Signals

- Many possible signals:
  - Hosts can observe E2E packet loss (e.g., TCP)
  - Hosts can observe E2E packet delay (e.g., Vegas, FAST)
  - Router can tell source of congestion (e.g., RED/ECN)
  - Router can tell source its allocation (e.g, XCP)

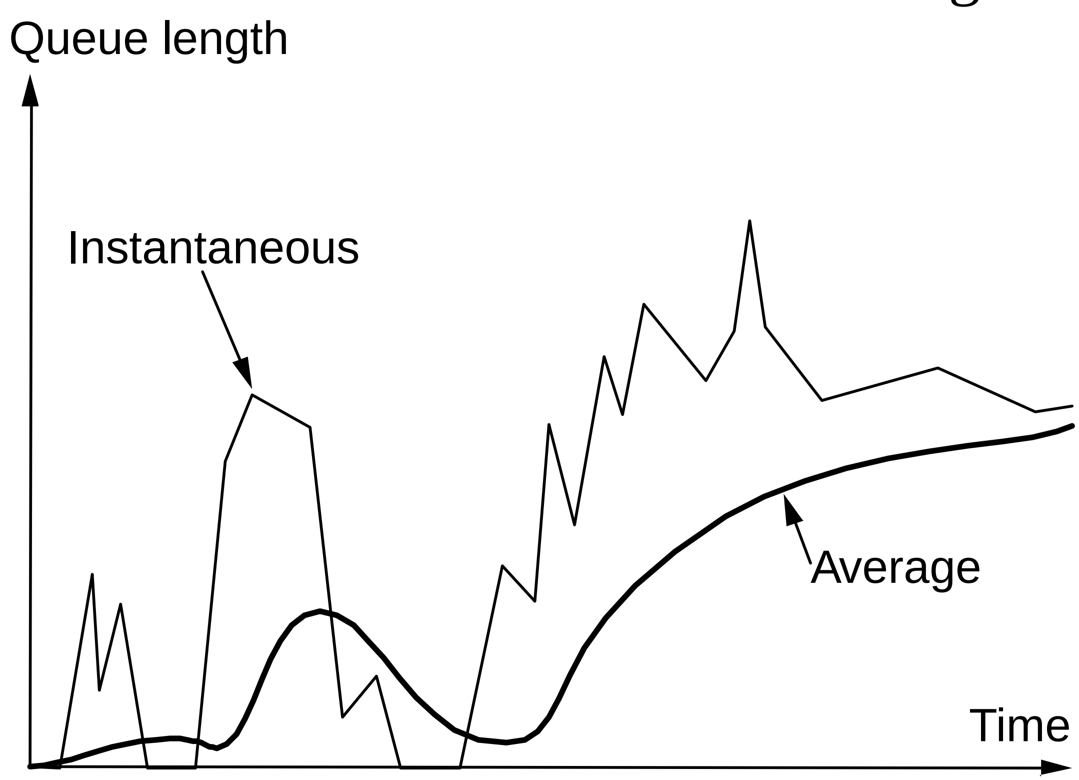
- Each has pros / cons and design implications

Avoidance versus Control

- Congestion control
  - Recover from congestion that is already degrading performance
- Congestion avoidance
  - Avoid congestion by slowing down at the onset
- Latter benefits from router support

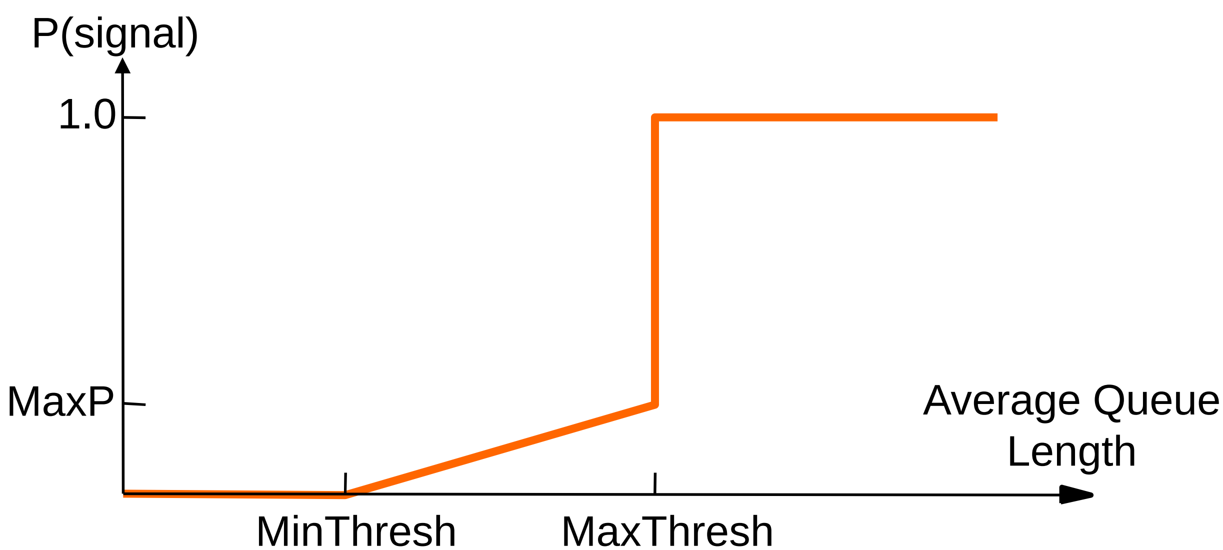
Detecting the onset of congestion

- Sustained overload causes queue to build and overflow
- Router can watch for an increase in the average delay



Random Early Detection (RED) routers

- Router sends “early” signal to source when avg. queue builds



- Probabilistically choose packet to signal; fast flows get more

RED signals

- Preferred/Future:
  - Set Explicit Congestion Notification bits in the IP packet header
  - Destination returns this signal to the source
  - Generates no extra packets at a time of congestion, signals reliably

- Deprecated/Present
  - Drop the packet; that is what pre-RED routers do
  - Source will get the hint
  - Paradox is that early loss can improve performance!
  - This is why RED tries to give each source only one signal

- In practice: not widely used, needs tuning to work well