CSE/EE 461 – Lecture 16

Bandwidth Allocation

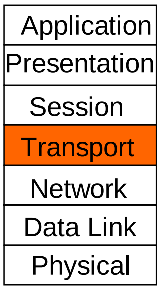
David Wetherall
djw@cs.washington.edu

Last Time

- The Transport Layer

- Focus
  - How do we decide when to retransmit?

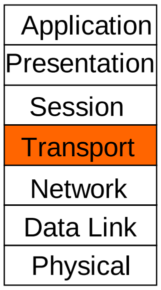
- Topics
  - Estimating RTTs
  - Karn/Partridge algorithm
  - Jacobson/Karels algorithm



Application
Presentation
Session
Transport
Network
Data Link
Physical

This Lecture

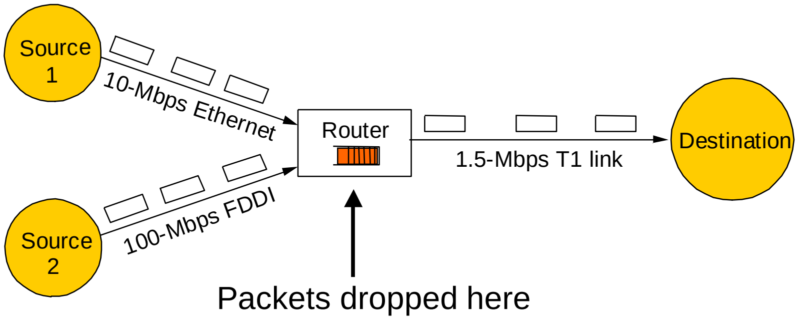
- The Transport Layer
- Focus
  - How do we share bandwidth?
- Topics
  - Congestion control
  - Fairness



Bandwidth Allocation

- How fast should the Web server send packets?
- Two big issues to solve!
- Congestion
  - sending too fast will cause packets to be lost in the network
- Fairness
  - different users should get their fair share of the bandwidth
- Often treated together (e.g. TCP) but needn’t be

Congestion

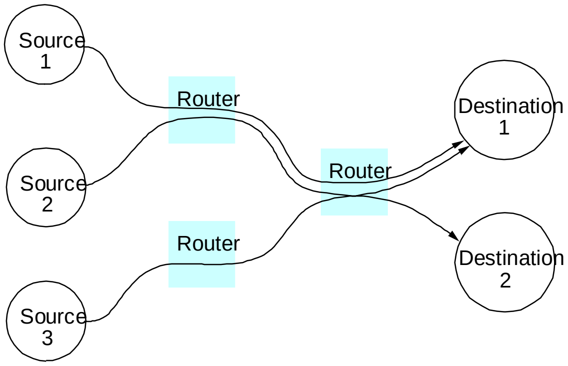


Packets dropped here

- 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

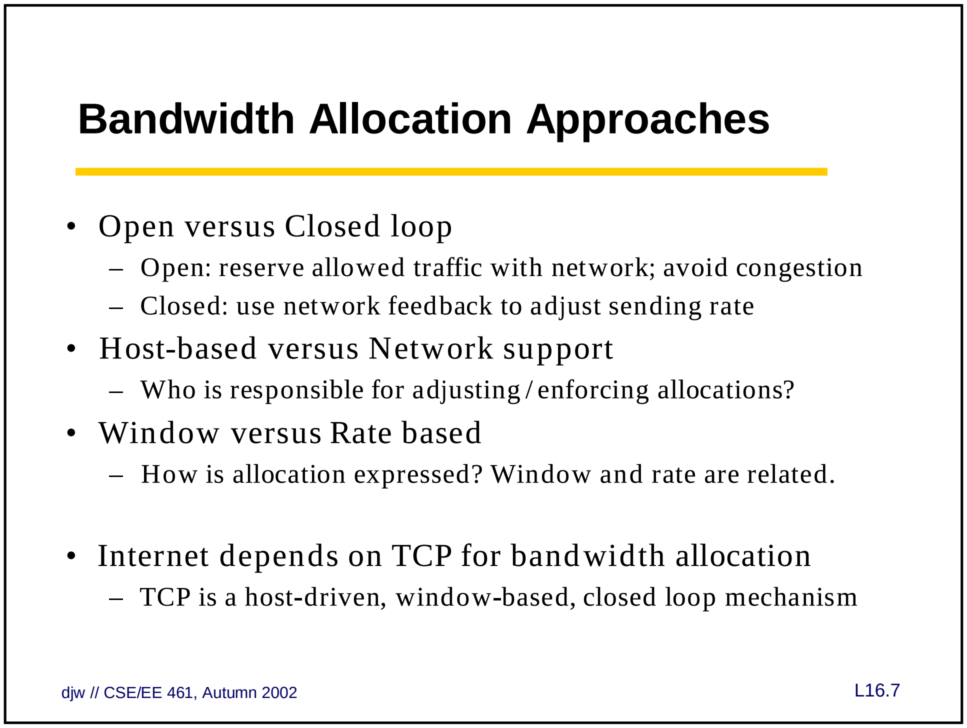
Chapter 6, Figure 1djw // CSE/EE 461, Autumn 2002

Fairness



- Each flow from a source to a destination should get an equal share of the bottleneck link … depends on paths and other traffic

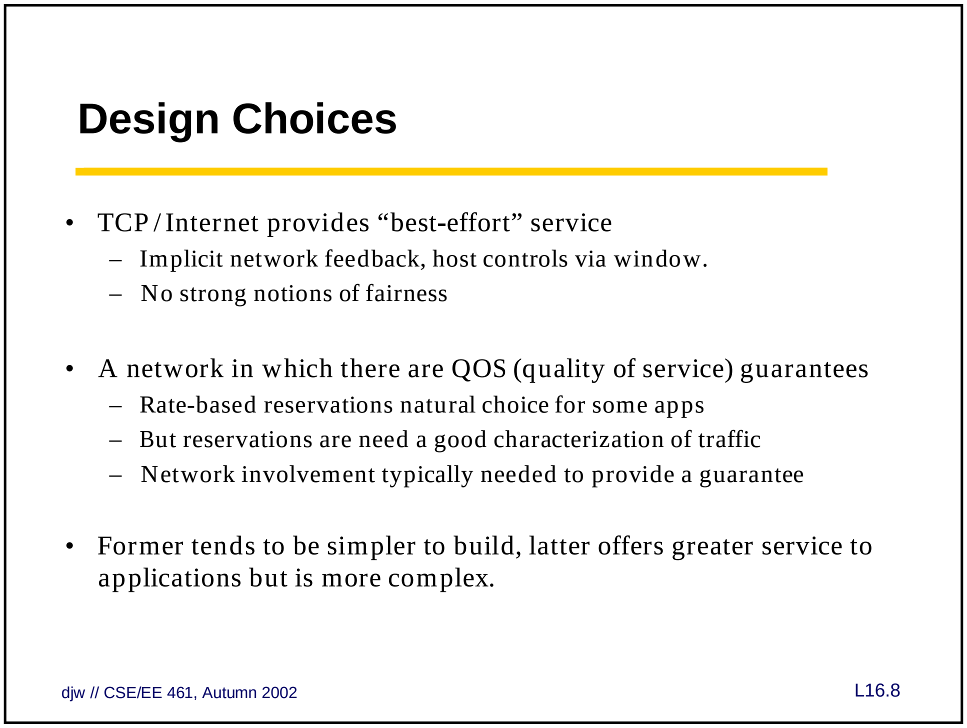
Chapter 6, Figure 2djw // CSE/EE 461, Autumn 2002



Bandwidth Allocation Approaches

- 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



Design Choices

- TCP/Internet provides “best-effort” service
  - Implicit network feedback, host controls via window.
  - No strong notions of fairness

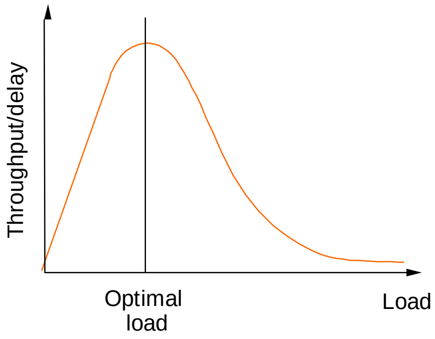
- A network in which there are QOS (quality of service) guarantees
  - Rate-based reservations natural choice for some apps
  - But reservations are need a good characterization of traffic
  - Network involvement typically needed to provide a guarantee

- Former tends to be simpler to build, latter offers greater service to applications but is more complex.

Evaluating Congestion Control

- Power = throughput / delay

- At low load, throughput goes up and delay remains small
- At moderate load, delay is increasing (queues) but throughput doesn’t grow much
- At high load, much loss and delay increases greatly due to retransmissions

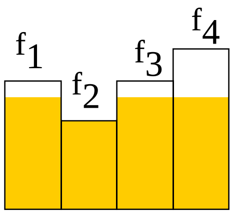


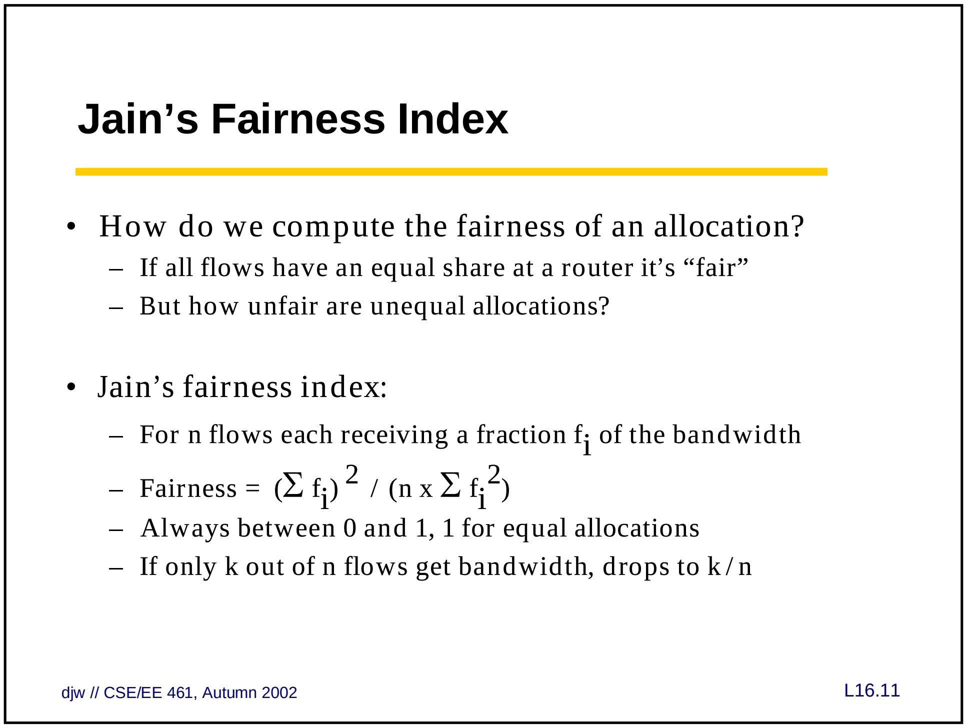
Evaluating Fairness

- First, need to define what is a fair allocation
  - Consider n flows, each wants a fraction f_i of the bandwidth

- Min-max fairness:
  - First satisfy all flows evenly up to the lowest f_i. Repeat with the remaining bandwidth.

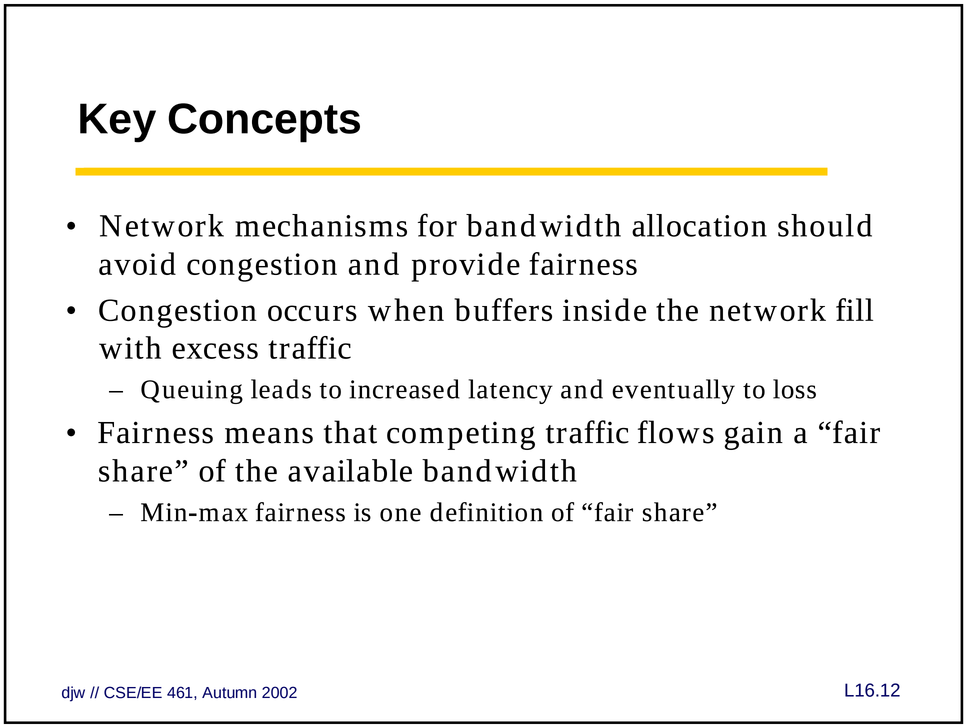
- Also proportional fairness
  - Depends on path length …





Jain’s Fairness Index

- How do we compute the fairness of an allocation?
  - If all flows have an equal share at a router it’s “fair”
  - But how unfair are unequal allocations?
- Jain’s fairness index:
  - For n flows each receiving a fraction f_i of the bandwidth
  - Fairness = $(\Sigma f_i)^2 / (n \text{ x } \Sigma f_i^2)$
  - Always between 0 and 1, 1 for equal allocations
  - If only k out of n flows get bandwidth, drops to k / n



Key Concepts

- Network mechanisms for bandwidth allocation should avoid congestion and provide fairness
- Congestion occurs when buffers inside the network fill with excess traffic
  - Queuing leads to increased latency and eventually to loss
- Fairness means that competing traffic flows gain a “fair share” of the available bandwidth
  - Min-max fairness is one definition of “fair share”