CS551
TCP Congestion Control
Bill Cheng

http://merlot.usc.edu/cs551-f12
Causes and Costs of Congestion

Queueing delays in router as packet arrival rate nears link capacity
- even if routers have infinite buffer space
  - costs: wasting bandwidth to forward unneeded copies

Retransmissions costs: (routers have finite buffer, so packet get dropped)
- routers have finite buffer (packets get dropped)
- retransmitted data eat up bandwidth
- when a packet is dropped along a path, the transmission capacity that was used at each of the upstream routers to forward that packet was wasted

The theory behind congestion control
- stability
- efficiency
Congestion

If both sources send at full speed, the router is overwhelmed

- **congestion collapse**: senders lose data from congestion
  and they resend, causing *more* congestion (can be self-reinforcing)

Other forms of congestion collapse:

- Retransmissions of large packets after loss of a single fragment
- Non-feedback controlled sources
Congestion Control and Avoidance

- A mechanism which:
  - Uses network resources efficiently
  - Preserves fair network resource allocation
  - Prevents or avoids collapse

- Congestion collapse is not just a theory
  - Has been frequently observed in many networks

```
throughput

knee  cliff
```

```
delay

knee  cliff
```

Load
Congestion Control vs. Flow Control

What does flow control do?
- avoids overrunning the receiver

What does congestion control do?
- avoids overrunning router buffers and saturating the network

What mechanism do they use?
- both use windows: $wnd$ for flow control and $cwnd$ for congestion control, actual window used is $\text{min}(wnd, cwnd)$
Congestion Control Goals

- Efficiency (maximize throughput or power [Ramakrishnan90a])
- Fairness [Ramakrishnan90a]
- Stability [Jacobson88a]
Fairness

- Should treat all users equally
  - But, defining fairness is hard... what is a user?
    - host, flow, person?
  - $n$ flows through a link, each flow should get $1/n$ bandwidth
    - what if their needs are different?

- Measuring fair allocations [Ramakrishnan90a]
  - In the absence of knowing requirements, assume a fair allocation means equal allocation
  - Jain and Chiu’s fairness index: $\left( \sum x_i \right)^2 / n \left( \sum x_i^2 \right)$
    - $x_i =$ throughput of flow $i$
    - $Ex$: fairness index = 1 if all $x_i$ are equal
    - $Ex$: fairness index = $k/n$ if $k$ out of $n$ flows are equal and other flows $(n-k)$ receives 0 throughput

- Other schemes, e.g., fair queueing [Demers89a]
Efficiency

- Want most throughput with low delay
  - System is most efficient at knee of curve
  - Power [Ramakrishnan90a]

\[
power = \frac{\text{throughput}^\alpha}{\text{delay}}
\]

- \(0 < \alpha < 1\), \(\alpha = 1\) results in power being maximized at the knee of the curve
- (others may say that the knee of the delay curve is at \(L_2\))
Avoidance or control?
- **Avoidance** keeps system at knee of curve
- But, to do that, need routers to send *accurate signals* (some feedback)
  - this is what ECN tries to accomplish
  - another possibility is to use rate (in the future)
- **Control** responds to loss after the fact

Sending host must adjust amount of data it puts in the network based on detected congestion
- TCP uses its window to do congestion control
  - but also avoidance, sort of
- But what’s the right strategy to increase/decrease window (slow start, congestion avoidance, exponential backoff)
How To Adjust Window in TCP?

When to increase/decrease cwnd?

A control theory problem
- Observe network
- Reduce window when congestion is perceived
- Increase window otherwise

Constraints:
- Efficiency
- Fairness
- Stability or convergence (too much oscillation is bad)
- Out-of-date information
  - RTT is fundamental limit to how quickly you can react
Formulation allows for the feedback signal:
- to change additively: $a_i(t)$
- to change multiplicatively: $b_i(t)$
- can consider feedback

What does TCP do?
- **AIMD: additive increase, multiplicative decrease**
- maximize stability: slow increase, fast decrease
Linear Control Example [Chiu89a]

![Diagram showing fairness line, optimal (efficient and fair) line, overload, underload, user 1's allocation $x_1$, and user 2's allocation $x_2$.]
Linear Control Example [Chiu89a]

user 1’s allocation $x_1$

user 2’s allocation $x_2$

fairness line

efficiency line (full utilization)

$(x_{10}, x_{20})$
Linear Control Example [Chiu89a]

- User 1’s allocation $x_1$
- User 2’s allocation $x_2$

Fairness line

Additive increase

Efficiency line (full utilization)

Point $(x_{10}, x_{20})$
Linear Control Example [Chiu89a]

Fairness line

Additive increase

Multiplicative increase

Efficiency line (full utilization)

(user 1’s allocation $x_1$, user 2’s allocation $x_2$)
Linear Control Example [Chiu89a]

- \( x_1 \): user 1's allocation
- \( x_2 \): user 2's allocation
- Fairness line
- Additive increase
- Multiplicative decrease
- Efficiency line (full utilization)

\( (x_{10}, x_{20}) \)
Linear Control Result [Chiu89a]

- Efficiency, fairness and distributedness imply that
  - Decrease must be multiplicative
  - Increase must be additive and may have a multiplicative factor
    - smoothness implies that multiplicative increase factor must be exactly 1
TCP Equally Shares Bandwidth On A Congested Link

A full bandwidth utilization line represents an equal bandwidth share between two TCP streams.
TCP Equally Shares Bandwidth On A Congested Link (Cont...)

A → B:
- no slow start
- congestion avoidance for both streams
- parallel to the 45-degree line

full bandwidth utilization line
equal bandwidth share

TCP stream 2 throughput
TCP stream 1 throughput

A
B
R
R
TCP Equally Shares Bandwidth On A Congested Link (Cont...)

TCP stream 1 throughput

TCP stream 2 throughput

R

full bandwidth utilization line

equal bandwidth share

B → C:
- packet loss, so ssthresh = cwnd/2
- C is half way between B and origin
TCP Equally Shares Bandwidth On A Congested Link (Cont...)

TCP stream 1 throughput

TCP stream 2 throughput

full bandwidth utilization line

equal bandwidth share

C → D:

congestion avoidance
TCP Equally Shares Bandwidth On A Congested Link (Cont...)

What if B → C does not go towards the origin? unfair?