TCP Congestion Control

- Closed-Loop Control
- Dynamically changes sender window size (congestion window size)

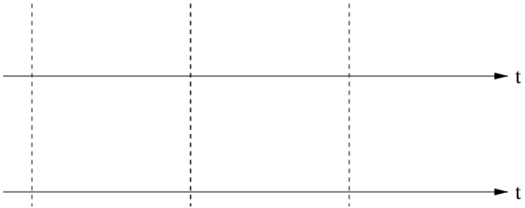
Issue

- Fairness

Flow Control - Congestion Control

- Flow Control: mechanism to prevent the sender from sending data when the receiver buffer is full.
- Congestion Control: mechanism to prevent congestion within the network.

Congestion Window - Transmission Rate



transmission rate $\leq$

MSS: Maximum Segment Size

TCP Congestion Control Algorithm (Tahoe)

Parameters

- *RcvWin*
- *threshold*
- w
- $CongWin = wMSS$
- $n = \min\{RcvWin, CongWin\}$

Init

- Set *threshold*
- Set $w = 1$

Slow Start

- As long as $w \leq threshold$, for every received ACK set

$$w = w + 1$$

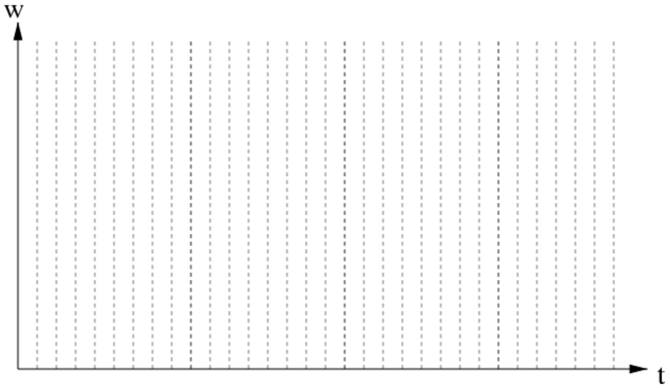
Congestion Avoidance

- When $w > threshold$, for every w ACK received set

$$w = w + 1$$

Loss

TCP Congestion Control Algorithm



TCP Congestion Control: Average Transmission Rate

Simple Model:

- Ignore “Slow Start” Phase
- Assume constant *threshold* $= W$

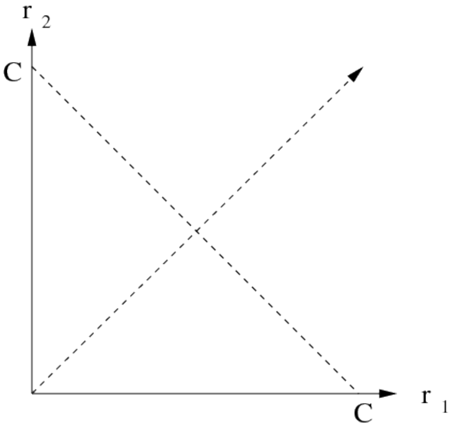
$$
\begin{aligned}
\text{Aver. Trans. Rate} &= \frac{1}{W/2+1} \sum_{k=W/2}^{W} \frac{kMSS}{RTT} \\
&= \frac{1}{W/2+1} \left[\frac{W(W+1)}{2} - \frac{(W/2-1)W/2}{2} \right] \frac{MSS}{RTT} \\
&= \frac{1}{W/2+1} \frac{0.75W^2 + 0.5W}{2} \frac{MSS}{RTT} \\
&\approx \frac{1}{W/2} \frac{0.75W^2}{2} \frac{MSS}{RTT} = \frac{0.75 \cdot W \cdot MSS}{RTT}
\end{aligned}
$$

TCP Congestion Control: Fairness

Simple Model:

- Ignore “Slow Start” Phase
- 2 Connection share one Link with Capacity *C*

Additive-Increase, Multiplicative-Decrease (AIMD) algorithm



Observations:

- Simple and Scalable
- Does not make any assumptions about Network Layer
- Max-Min Fair

Drawbacks

- Based on Packet Loss
- No QoS-Guarantees
- Is Max-Min Fairness what we want?
- Assumes User Cooperation
- Vulnerable to UDP sessions
- Average Transmission Rate dependent on RTT