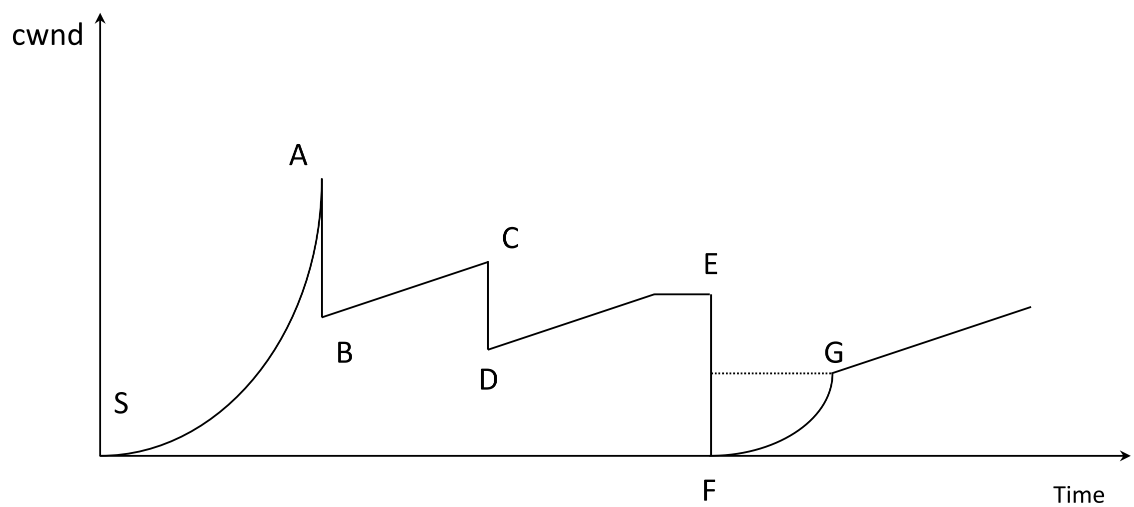
# Homework 3

*Due: 13 April, 2012, 4pm*

## Problem 1 - TCP

a. The figure below shows the evolution of the congestion window of a TCP Reno connection.




(a) What is happening between points S-A (i.e. what mode is TCP in)? What happens at point A to cause the change in window size? [1 pt]

(b) What happens at point E to cause the change in window size? How is this different from the event at point A? [1 pt]

b. We saw that an approximation for the average throughput of TCP is given by

$$T = \frac{1.22MSS}{RTT\sqrt{L}},$$

where $L$ is the loss rate. For a path with 1,000 byte segments, RTT of 200ms, what is the loss rate if we want to saturate a link of 200Mbps? What about a link of 5Gbps? [1 pt]

c. Suppose you have two TCP connections sharing a bottleneck link on the network: both connection A and connection B have a RTT of 100ms but A has a MSS of 1500 bytes and B has a MSS of 750 bytes. Both operating in TCP Reno congestion avoidance mode and are adding 1 MSS per RTT. In the long run, will they reach the same throughput? Justify your answer using a Chiu-Jain phase plot. Assume the flows start at a point well below the full utilization of the link, and that they both reduce their rates by half whenever they cross the full utilization line. Hint: draw what happens at every increment of 100ms. [2 pts]

## Problem 2 - TCP and HTTP

Suppose that a Web browser has to download 30 objects from the same server to properly display a page. Assume that these objects are all 15KB long and that the MSS for the connection is 1KB. The communication between the client and the server has to go through a bottleneck link that has a total bandwidth of 1MB/s, and there is one TCP flow already present in this bottleneck link. Assume that the RTT for your flows, as well as for the other flow already there, is 100ms.

We are going to assume an idealized version of TCP, called "TCP Providence", that has only one phase after regular connection establishment that sends data a constant rate. No slow start phase is necessary because you can assume TCP Providence is able to magically know the size of the `cwnd` that will lead to the "fair" sharing of the bottleneck link. In other words, assume that TCP will start each connection with a window equal to the "fair" size of the congestion window. You can also ignore the closing of the connections.

a. If yours is the only flow in the bottleneck link, what would be the size of the congestion window that would maximally utilize the link? What would the window size be if you added the other flow mentioned above? (Assume this extra flow is going to be there for all subsequent questions). [2 pts]

b. If the browser can open just one concurrent TCP connection to the server, using HTTP/1.0, how long would it take to transfer all of the 30 objects? [1 pt]

c. What if the browser adds HTTP/1.1 pipelining? (Assume all of the 30 requests can fit in one segment), how long does it take now? [1 pt]

d. Now the browser gets greedy, and decides to open 30 parallel connections to the server, requesting one object on each connection. How long will it take to transfer all files? [1 pt]

## Problem 3 - DNS

a. DNS packets make up a significant portion of Internet traffic. Give two techniques that the DNS system employs that allows it to scale to serve the entire Internet. Are there downsides to either of these techniques? [2 pts]

b. Explain how SOPA and DNSSEC conflict. [1 pt]