CSCI 415 Data Communication Networks
Homework 8
05/14/08

Saad Mneimneh
Visiting Professor
Hunter College of CUNY

Problem 1: MIMD (optional)
This problem is inspired by a question asked in class. Consider an congestion control algorithm that always uses slow start to increase the window size. Therefore:

$$on\ ACK:\ cwnd = cwnd + 1$$
$$on\ timeout:\ cwnd = cwnd/2$$

(a) Under what circumstances is the fast retransmit mechanism not effective?

(b) Suppose that the timeout is equal to $\alpha \times RTT$ (in practice α could be as large as 20). Show by constructing an example that the effective throughput of this congestion control algorithm could be as low as (assuming a unit of bandwidth)

$$\frac{2}{\alpha + 1}$$

Problem 2: Random Early Detection
Consider a RED gateway with $MaxP = 0.02$, and with an average queue length halfway between the two thresholds. Recall that p is computed as $\frac{Q_{avg} - Q_{min}}{Q_{max} - Q_{min}} MaxP$ and $p_{count} = \frac{p}{1 - count \times p}$, where *count* is the number of packets that are queued since the last drop while $Q_{min} \le Q_{avg} \le Q_{max}$.

(a) Find the drop probability p_{count} for $count = 1$ and $count = 50$.

(b) Calculate the probability that none of the first 50 packets are dropped. Note that this is $(1 - p_1) \times \ldots \times (1 - p_{50})$.

Problem 3: Fair Queueing
Consider a router that is managing three flows, on which packets of constant size arrive at the following real times:

flow A: 1, 2, 4, 6, 7, 9, 10

flow B: 2, 6, 8, 11, 12, 15

flow C: 1, 2, 3, 5, 6, 7, 8

All three flows share the same outbound link, on which the router can transmit one packet per time unit (therefore, we can think of every packet having a length of 1, and the router having a service rate $c = 1$). Assume there is an infinite amount of buffer space.

(a) For each packet, give the real time when it is transmitted by the router if the router is an ideal round robin system.

(b) For each packet, give the real time when it is transmitted by the router if the router implements Fair Queueing. Ties are to be resolved in the order A, B, C.

PS: there will be quite an amount of bookkeeping. So try to keep track of information using a table to compute $V(t)$, a_i^k, S_i^k, F_i^k, and which packets are still in queues, etc...

t	arrivals	$V(t)$	#flows(ideal sys)	F_p	t_{next}	sent	A's queue	B's queue	C's queue