HW #2

ELEN E6761 - Computer Communication Networks — Due 9/13/2000, 5pm
Fall 2000 — Prof. Rubenstein

Please turn your homework into Vassilis Stachtos' mailbox, E2 (oustide of the Electrical Engineering office, fax it by 5pm, or make some other arrangement with Vassilis. On-campus students can fax to Professor Rubenstein at (212) 932-9421 (remember to include your name and that the fax is for Professor Rubenstein). CVN students can fax as above, or fax to the CVN fax number. LATE ASSIGNMENTS WILL NOT BE ACCEPTED!! Note: don't wait until the last minute to fax - only one can go through at a time.

1. Given a linespeed of l bits / second, a packet size of k bits, a round trip time of R, and a window size of W, what is the rate of communication? Assume that there is no packet loss and that R includes the time it takes the receiver to receive, process and acknowledge the data packet, and also includes the time it takes the sender to receive and process the acknowledgment. However, it does not include the time that it takes the sender to place the data packet onto the wire.

2. Consider a reliable data transfer protocol that transfers data using a window of size w. Assume that packets can be lost, but cannot be reordered. Show that the minimum number of sequence numbers that the protocol must utilize to guarantee error-free transfer when the transfer is performed in the style of

   (a) Selective Repeat is $2w$.

   (b) Go-Back-N (assume the receiver discards any packets that cause a gap in the sequence numbering) is $w + 1$.

   (c) 10% EXTRA CREDIT: Assume now that packet ordering can be shifted by up to n, i.e., the jth packet transmitted by the sender might arrive after the $j + n - 1$st packet. How does this effect the required minimum size of the sequence numbering for both cases above? FORMAL PROOF REQUIRED TO RECEIVE FULL CREDIT!

3. Assume the bounds on round trip time (RTT) delay between a pair of hosts, A and B, lie within the interval, $[r_1, r_2]$, and that samples are drawn from a distribution that is independent of time (i.e., the correlation between successive sample values is not a function of the time that elapses between the samples). A protocol is designed with the assumption that the estimate for the round trip time does not vary more rapidly than 1 sec/sec (i.e., over any T units in time, the RTT estimate should not change more than T). Assume the protocol uses an exponentially decaying average with parameter α that indicates the weight of the new sample (i.e., $A((n+1)T) = (1-\alpha)A(nT) + \alpha S_{(n+1)T}$ where S_t is the sample taken at time t, and $A(t)$ is the average computed at time t.) Note that the average varies less with time as the weight, α, is reduced, or as the sampling period, T, is increased. To estimate the round trip time and meet the requirements of the protocol:

   (a) given that the protocol samples the RTT every T seconds, what is the maximum value for α such that the average meets protocol requirements? Explain.

   (b) what is the minimum sampling period as a function of α so that the average meets protocol requirements? Explain.

   (c) Describe the difference in how the exponentially weighted average changes with time depending on whether α is varied versus if the sampling period, T, is varied.

4. In class (Lecture 4 notes, slide 36), we showed that when flows traverse two links, a situation can arise where increasing throughput rates can in fact lead to a drop in goodput rates as a result of congestion. Set up (but do not solve) the system of equations for flows that traverse n hops each link is the ith hop for one flow for all $0 < i \leq n$. For instance, in Figure 1, $n = 5$ - only one flow is shown, there must be 5 other flows [6 flows in total] , 1 starting at each link). Assume every link L has capacity C_L and

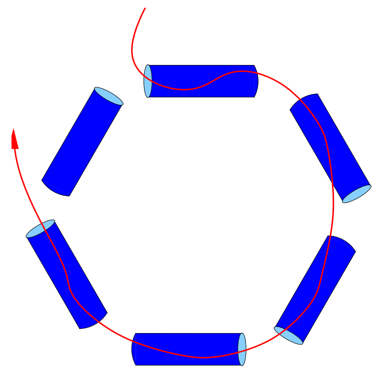

Figure 1: A 6 hop network with one connection traversing 5 of 6links

let D_L be the rate at which packets (or more appropriately, bits or fluid) is dropped on every link L (the drop rate is the same due to symmetry of the network in the example.

Helpful hint: The easiest way to proceed is to consider what happens at a single link. Let λ_i equal the rate at which flow enters the link i hops from its source, and let λ_n equal the rate at which a flow exits the link after its nth hop (in the example on p. 36, λ_1 would be relabelled as λ_0, λ_p would be relabeled as λ_1, and λ_x would be relabeled as λ_2). Certain equations will have summations: be specific (and careful) about the range that the summation iterates over.

5. A sender uses a leaky-bucket mechanism with rate λ and bucket size W to transmit packets into the network. The sender also maintains a window of size W so that the number of sequence numbers in transit at any time is at most W (i.e., more than W packets can appear in the network at a time but then there must be copies of the same packet). Let $D(\lambda)$ be the average delay between the time a packet is transmitted and the time the acknowledgment of its receipt reaches the sender when the sender is transmitting at rate λ. Let p be the probability of packet loss, and assume that if a packet is lost, the sender does not find out about the lost packet for this same amount of time (i.e., it cannot retransmit until time $D(\lambda)$ after the previous transmission of the packet.

   (a) Assume $D(\lambda) = A\lambda + B$ and $p = 0$. What is the maximum rate at which the sender can transmit?

   (b) Suppose $p > 0$ and $D(\lambda) = A'\lambda + B$ and a selective repeat protocol is used. Give an upper bound on the throughput and goodput. Why is this just an upper bound?

   (c) Suppose the protocol used is Go-Back-N. Does the throughput upper bound still hold? If so, is it a tighter or looser upper bound? What about for the goodput? Explain all answers.

6. Consider an M/M/1/K queue into which packets arrive at rate λ and are serviced at rate μ (assume for simplicity that $\mu \neq \lambda$ and a queue size of K. Consider a queueing discipline in which packets arrive marked as priority with probability p or else they arrive as non-priority. Furthermore, assume that when the processor selects its next packet, it always selects a priority packet if one is available, and that the system is pre-emptive (i.e. it postpones processing of a non-priority packet as soon as there is a priority packet in the system). Furthermore, if the queue is full when a priority packet arrives, then a non-priority packet is discarded to make room for the priority packet.

   (a) What is the loss rate of the system? (Hint: consider the liklihood that the queue is full and apply PASTA)

   (b) What is the expected delay experienced by a priority packet?

   (c) Assume K is infinite and that $\mu > \lambda$. What is the expected delay experienced by a non-priority packet? (HINT: use the expected value of an "average" packet's delay and the expected value of a high priority packet's delay)

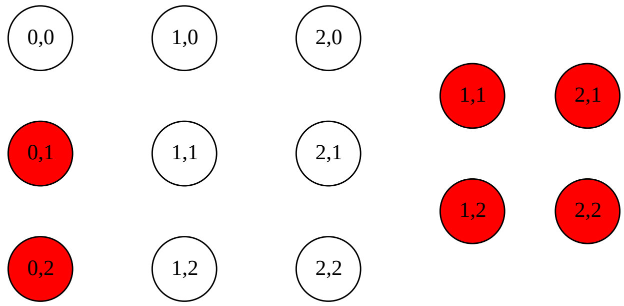

Figure 2: A Markov model for a non-preemptive M/M/1/K queue

(d) Assume now that the queue is non-preemptive. Fill in the transitions and their probabilities for Figure 2 (assume that $K > 2$, but that not all states are shown in the Figure). Shaded states are those in which a low-priority packet is currently being serviced.

7. Draw the trie structure that maps addresses to the interface corresponding to the longest matching prefix in the table below:

prefix	interface
0	1
1	1
10	2
000	3
100	1
1001	2
0100	1

8. In this problem, you will build various components of the trie-maintenance algorithm:

(a) Sometimes there are redundant entries in a lookup table. For instance, consider the following lookup table:

prefix	interface
0	1
1	1
10	1
000	3
0000	3
0001	3
0100	1

Entries 0000 and 0001 are unnecessary since the entry 000 covers these two cases. It is therefore a waste of memory for the lookup table to contain entries 0000 and 0001. Devise an algorithm called algorithm_clean() that, given a trie, filters out the unncessessary entries (e.g., in this example, 0000 and 0001 would be removed.) Hint: construct a function that can be called by itself recursively.

prefix	interface
0	1
1	1
10	1
0000	3
00010	3
00011	3
0100	1

(b) Consider the prefix lookup table above. Prefixes 0000, 00010, and 00011 can safely be replaced with the single prefix 000. Briefly describe how you would modify your algorithm to perform such replacements.