HW #6

ELEN E4710 - Intro to Network Engineering — Due 11/19/2003
Fall 2003 — Prof. Rubenstein

Homework must be turned in at the beginning of class on the due date indicated above. CVN students have one additional day. Late assignments will not be accepted.

1. A set of flows $f_1, \cdots f_n$ transmit packets through a router. The router operates on a round-by-round basis, processing at most 1 packet per round. Each flow delivers at most one packet to the router during each round.

   (a) Assume **priority queuing** is used, where flow i is given priority over all flows $j > i$. Assume the router is bufferless and must discard any packets that it receives in round k that it cannot process during round k. Each round, each flow transmits a packet to the router during that round with probability p. What is the expected number of packets processed by the router belonging to flow i after m rounds?

   (b) Suppose **virtual clock** is used instead and the router can store an infinite number of packets, but can still only process at most one packet per round. Assume there are 2 flows, where flow 2 gets served first whenever the two flows' virtual clocks have the same value. If flow 1 sends a packet every round and the flow 2 sends a packet in a round with probability p, what is the expected number of rounds needed to process m packets from flow 1. (Hint: let X_i be a r.v. that equals 1 if flow 1 finishes in round i and 0 otherwise. Leave your answer expressed as a sum).

2. A router that can process two packets simultaneously and buffer a third is used to transmit two flows. Flow 1's packets arrive at the router at rate λ_1, flow 2's packets arrive at rate λ_2, and processing a packet (from either flow) is performed at rate μ. Note that when two packets are being processed by the router simultaneously, each one is processed at rate μ. When both processors are busy, arriving packets from flow 2 are *always* discarded, and arriving packets from flow 1 are discarded unless the 1-packet buffer is empty, in which case the packet from flow 1 is stored there for processing once a processor becomes available.

   Assuming that all arrival processes are Poisson and packet processing times are exponential, draw a Markov model that describes the system.

   Hint: to accurately describe the system, your model needs at least 9 states.

3. Construct a FSM for a router that implements weighted fair queueing for two flows f_1 and f_2 where $w_1 = 5, w_2 = 2$, and the slack is $1/2$ for both flows (where the processing of a packet from f_1 adds $1/5$ to its virtual clock, and the processing of a packet from f_2 adds $1/2$ to its virtual clock). Your state machine can depend on the following functions and events to simplify its design:

   - $IE()$: the queue is empty (prior to an arrival)
   - $H(i)$: $H(i)$ equals 0 when no packets from f_i are queued in the system, and equals 1 otherwise
   - $D()$: triggered when the router completes processing its current packet.
   - $A(i)$: triggered when an arrival to the router from f_i occurs (only needs to be used when the router is not processing any packets).

   With these functions and events, you should build your FSM so that it indicates clearly whose packet should be processed next whenever such a decision needs to be made. (Hint: each state should indicate the current difference between the two flow's clocks)

4. Construct a Markov model for a round-robin queueing system that can store up to 3 packets for processing. Assume there are two flows in the system, where both flow's packets are processed at rate μ, and flow f_i's packets arrive at rate λ_i. Label transistions with their transition probabilities.

5. Consider a queue that can store up to k packets, as we considered in class. We consider a modified version of discrete birth-death process we considered in class where the queue, when empty, processes "pretend" packets. When there are packets in the system, then with probability p, the next event is a packet arrival, which is dropped if the number of packets in the queue is already k. With probability $1 - p$, the next event is the processing of a packet. If the number of packets in the queue was greater than 0, then after processing, the number of packets is decreased by 1. If the number of packets was equal to 0, then the processing was of a "pretend" packet, and the size of the queue stays at 0.

   Define $S(n)$ to equal the number of packets in the system after the nth event (arrival or processing of a packet). Here, an event is a packet arrival or a service completion, including the completion of pretend packets.

   (a) What are $\Pr(S(n) = 1 | S(n-1) = 0)$ and $\Pr(S(n) = 0 | S(n-1) = 0)$?

   (b) Construct the Markov model for this queueing system.

   (c) Solve for the steady-state distribution, $\pi_i = \lim_{n\to\infty} \Pr(S(n) = i)$ for all $i, 0 \le i \le k$.

   (d) Solve for π_i as $k \to \infty$.

   (e) π_i is different than the value computed using the Markov model formulated in class. Suppose we consider a continuous time version of this problem where packets arrive at the queue with rate λ and are processed at rate μ, where $p = \lambda/(\lambda + \mu)$. Which set of π_i would match the π_i for the continuous version of this Markov Model? Why?