Coursework 2

Network Performance—DJW—2005

1. Everyone answered correctly.

2. The question asks you to explain your code. Explanations ranged from passably good to terrible. It is not enough to hand in your code with a few comments. The explanation need be no longer than a few paragraphs, but it should explain your simulation approach, and it should have enough detail for a skilled programmer to duplicate your method.

Most students simulated the Markov process directly, i.e. they simulated the process (T_t, V_t). At each state, they typically simulated four exponential random variables, corresponding to the four possible transitions (new voice call, new video call, existing voice call ends, existing video call ends), found which was the minimum, and moved the Markov chain to the corresponding next state. Note that if the minimum of the four exponential random variables is m then m is the time spent in state (T_t, V_t). Many students mistakenly associated the duration m with the subsequent state.

One student produced a very elegant solution, in which he/she calculated the Q-matrix then wrote a general-purpose Markov process simulator which takes as its input a Q-matrix.

Some students chose to simulate each call explicitly, i.e. they treated the circuits as a container, and created call objects which could occupy slots in this container. This is valid, but it will run slower. These students also ran a slotted time system, i.e. simulated time in discrete steps, and associated each event (call arrivals and departures) with one of the time slots. If you do this, you must comment intelligently on how to choose the slot duration: too coarse and the simulator won't be accurate (and you may run into problems if several events are scheduled for the same slot); too fine and the simulation will be inefficient. You can judge this by looking at the highest transition rates in the Markov chain. (Left as an exercise.)

In a slotted time simulator, you have two choices about how to model call arrivals. You can generate exponential random variables to say when the next call is due, and use a 'lock' to make sure no more calls are generated in the intervening time; or you can flip a biased coin in every single time slot to decide whether or not a new call arrives. See the lecture notes on slotted time model of ALOHA for details.

Several students used threads to run multiple simulations concurrently. I see no benefit in this—all you'll achieve is to make the CPU/cache churn; you might as well run them one after the other. One student ran his entire simulation using threads, i.e. each call was a thread, and call durations were implemented by `sleep()` statements. This is interesting and unusual approach, but I suspect it's very inefficient and susceptible to thread-scheduling idiosyncracies.

You have several choices about what to measure. For example, to measure voice call blocking probability, you could count the number of blocked voice calls N_B, the number of attempted voice calls N_A, and calculate N_B/N_A. Or you could count the total amount of time the system spends in states in which an incoming voice call would be blocked T_B, and the total amount of simulation time T, and calculate T_B/T. Markov theory says that the two methods should agree, if you run your simulation long enough. Any students who spotted this choice and commented on it gained bonus marks.

The question asks you to plot "total number of occupied circuits *as a function of* time". This means that time is the x axis, and number of occupied circuits is the y axis. You are not being asked to plot the amount of time spent in each state (i.e. a graph with time on the y axis and state on the x axis).

3. Those who attempted this got it right (apart from minor numerical problems). The point of this question was that it can validate your simulation answer from Question 2. You should be able to spot that the answers to the two questions should be the same; and if the answers are different I expect you to comment intelligently on the reason for the difference.

4. There were marks available for four components: deciding an objective, articulating your objective, executing your objective, and obtaining a reasonable answer. I left it to you to decide the objective, and so I expect you to explain your choice. Students who spotted that there was a choice of objective and discussed this gained bonus marks.

What are sensible objectives? A network operator would probably like to maximize revenue. One can also consider fairness as an objective (video calls cost six times more, but they take four times the capacity, so they deserve 4/6 the blocking probability of voice calls.)

Everyone chose to answer this question by simulation. In my opinion it is easier to answer by calculation, as in Question 3; though simulation is entirely acceptable.