Midterm Exam - Solutions

Date: 26 November, 2012

Problem 1 (20 points): Consider the following discrete-time Markov chains.

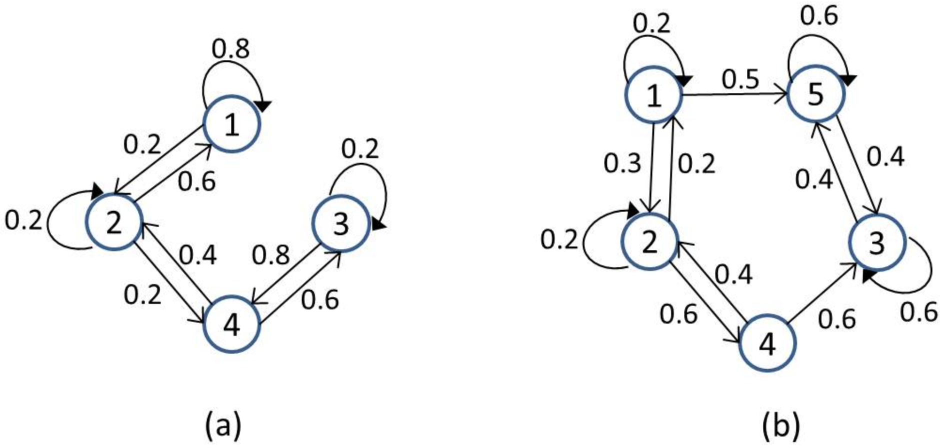

Figure 1:

For each of them answer the following questions:

1. Is the chain irreducible?
2. How many state classes does it have, and what are the states in each class (transient, positive recurrent or null-recurrent)?
3. Is the chain time-reversible?
4. If we start at state 1, find the limiting probability of $\lim_{n\to\infty} P_{1i}^{(n)}$ (i.e. the probability that we are at state $i = 1, 2, 3, 4$ after a large number of steps) (if the chain is ergodic, this is the stationary probability π_i.)

Solution:

Chain (a)
1) The chain is irreducible because every state is reachable by every other.
2) Since it's irreducible, it has only 1 class. All states are positive-recurrent as the chain is finite.
3) The chain is time-reversible. You can either argue about it by saying that when you go from state i to $i+1$, you can only come back to state i via $i+1$. Or you can try (below) to see if the stationary distribution satisfies the local balance equations.
4) Since the chain is time-reversible, we write the local balance equations along with the normalization equation.

$$\begin{aligned}
0.2\pi_1 &= 0.6\pi_2 \Rightarrow \pi_2 = \frac{1}{3}\pi_1 && (1)\\
0.2\pi_2 &= 0.4\pi_4 \Rightarrow \pi_4 = \frac{1}{2}\pi_2 = \frac{1}{6}\pi_1 && (2)\\
0.6\pi_4 &= 0.8\pi_3 \Rightarrow \pi_3 = \frac{3}{4}\pi_3 = \frac{1}{8}\pi_1 && (3)\\
\pi_1 + \pi_2 + \pi_3 + \pi_4 &= 1 \Rightarrow (1 + \frac{1}{3} + \frac{1}{6} + \frac{1}{8})\pi_1 = 1 \Rightarrow && (4)
\end{aligned}$$

This gives the stationary vector $\{\frac{24}{39}, \frac{8}{39}, \frac{3}{39}, \frac{4}{39}\}$
Chain (b)
1) The chain is not irreducible. Cannot go from 3 to 1, for example.
2) There are two classes: $\{1, 2, 4\}$ which are transient states and $\{3, 5\}$ which are positive-recurrent.
3) The chain is not time-reversible (as it's not even irreducible).
4) States 1, 2, and 4 are transient so $lim_{n\to\infty}P_{i1}^{(n)} = lim_{n\to\infty}P_{i2}^{(n)} = lim_{n\to\infty}P_{i4}^{(n)}0$.

We can therefore focus on the positive recurrent states 3 and 5. Using local balance we can immediately see that $\pi_3 = \pi_5 = 0.5$

Problem 2 (10 points): Given that power is expensive, it is common practice to leave servers ON only when they are being used, and turn them off whenever they are not in use. Assume that the following power-aware algorithm is used: When a job arrives, it instantly turns on a fresh server (assume zero setup cost). When the job completes service, it instantly turns off that server. Assume that there is always a server available for every job; i.e., there is no queueing. Your goal is to derive the average rate at which power is used in our system. Assume that when a server is on, it consumes power at a rate of $P = 240$ Watts. Assume $\lambda = 10$ jobs arrive per second, and that the service requirement of jobs is Exponentially-distributed with mean 5 seconds.

Solution: We know that $\lambda = 10 jobs/sec$ and $E[T] = E[S] = 5sec$, since theres never any queueing in this system. Hence, applying Littles Law, we have that:

$$E[N] = \lambda E[T] = 50$$

This says that there are 50 jobs in the system on average, or 50 servers in operation.

But since each server uses 240 Watts, the average power consumption is $50 \times 240 = 12000$ Watts.

Problem 3 (15 points): Consider N nodes in a wireless LAN with a single base station. Each of the N nodes has exactly 1 packet to transmit to the base station. To coordinate with each other and avoid collisions the following simple protocol is used:

- time is slotted; every node transmits in each slot with probability p.
- when exactly one node transmits in a slot, there is a success, and that node will never have to transmit again.
- when more than one node transmits in the same slot there is a collision; each of the nodes colliding will simply retry with the same probability p, until they succeed.

Question 1 (10 points): What is the expected number of slots until all nodes transmit successfully?

Question 2 (5 points) : What is the maximum value of p which guarantees that all nodes will eventually manage to transmit successfully?

Solution:

Question 1: We can model this problem with a markov chain with N states, with state k corresponding to k nodes not having transmitted successfully yet. Transitions are only possible from state k to state $k-1$ (for each k). When at state k, each node will transmit in the next slot, independently, with probability p. The probability of having exactly one successful transmission is

$$p_{k,k-1} = kp(1-p)^{k-1},$$

while with probability $p_{k,k} = 1 - kp(1-p)^k$ either no transmission or a collision occurs.

Hence, the transition time from state k to $k-1$ is a geometric random variable with probability $p_{k,k-1}$, and the expected transition time is

$$\frac{1}{kp(1-p)^{k-1}}.$$

The total delay to go from state N to state 0 (i.e. all nodes have transmitted successfully) is then

$$\sum_{k=1}^{N} \frac{1}{kp(1-p)^{k-1}}$$

Question 2: As long as the probability p is not equal to 1 (in which case, every node tries at every slot, which always results in a collision), all nodes will transmit successfully in finite time. This is also easy to see from the above sum, which for $0 < p < 1$ is just a finite sum (of finite numbers).

Problem 4 (20 points): Figure 2 shows a queueing network with 3 servers. All servers have exponentially-distributed service times. The service rate at server i is μ_i as shown. Outside arrivals occur according to a Poisson Process with rate 1 packets/sec into server 1. The scheduling policy at each server is FCFS. The edges of the network indicate routing probabilities.

1. **Question 1 (10 points) :** What is $E[T]$, the mean response time for a packet entering this network from outside?

2. **Question 2 (10 points) :** Your boss asks you to find a way to improve the performance ($E[T]$) of the system. You consider the following three options:

   - **Option 1:** double the speed of server 3 (i.e. $\mu_3 = 8$).
   - **Option 2:** double the speed of server 2 (i.e. $\mu_2 = 10$).
   - **Option 3:** add a second server in parallel with server 3 (with the same rate $\mu_4 = 4$ pkts/sec) as shown in Figure 3 and split the load equally.

     Rank the three options from most improvement to least improvement.

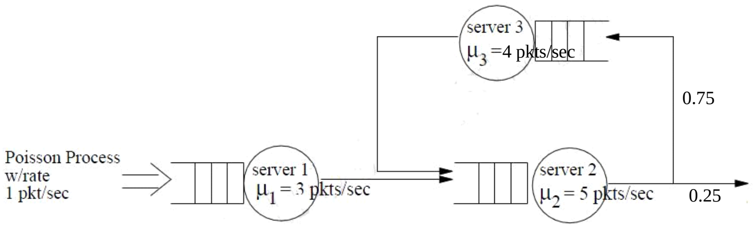

Figure 2:

Solution:
Question 1: We first need to find the input rates for each of the 3 queues.

$$\begin{aligned}
\lambda_1 &= 1 \\
\lambda_2 &= \lambda_1 + 0.75\lambda_2 \\
\lambda_3 &= 0.75\lambda_2.
\end{aligned}$$

which gives

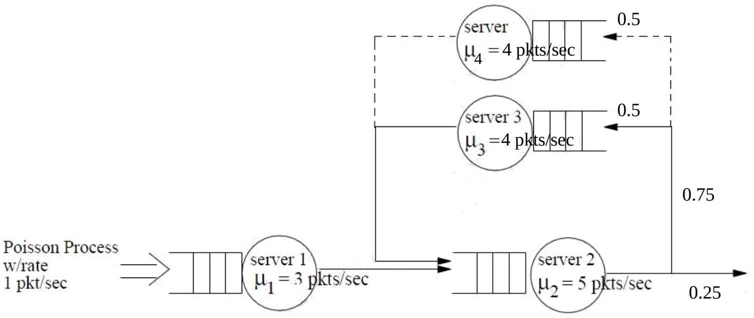

Figure 3: Option 3

$$\begin{aligned}
\lambda_1 &= 1 \Rightarrow \rho_1 = \frac{1}{3} \\
\lambda_2 &= 4 \Rightarrow \rho_2 = \frac{4}{5} \\
\lambda_3 &= 3 \Rightarrow \rho_3 = \frac{3}{4}.
\end{aligned}$$

We can thus derive the expected number of customers in each queue using the basic M/M/1 equation for $E[N_i]$

$$E[N_1] = \frac{\rho_1}{1-\rho_1} = \frac{1}{2}, E[N_2] = 4, E[N_3] = 3.$$

Thus, the total number of customers in the system is $E[N] = \sum_i E[N_i] = 7.5$, and since customers can only enter from queue 1, we can use Little's Law to get

$$E[T] = \frac{E[N]}{\lambda_1} = 7.5sec$$

Question 2: We can perform the exact same analysis for the 3 new options (note that in option 3, traffic leaving server 2 and fed back with prob 0.75 is now split equally into the two servers, 3 and 4). This gives us $E[T^{(1)}] = 5.1, E[T^{(2)}] = 4.167, E[T^{(3)}] = 5.7$, which makes option 2 the best one.

We could have already guessed that though, since server 2 is the *bottleneck* in the original system, so improving it should have the best impact in the overall system performance.

Problem 5 (20 points): Customers arrive in a usual M/M/1 system, with an arrival rate λ and service rate μ. However, customers in the queue are impatient: Each customer waiting in the queue will abandon the system without receiving service with a rate γ (i.e. each user will abandon the system with probability $\gamma\Delta t + o(\Delta t)$ in any interval of duration Δt).

- **Question 1:** Draw the CTMC for the above system.
- **Question 2:** Derive the stationary probabilities π_i for this chain.
- **Question 3:** For what values of input rate λ is the above queue stable?
- **Question 4:** What is the total rate of *lost* customers per time unit?
- **Question 5:** What is the throughput of the system?

Solution:

Question 1: When at state k, 1 node is in service and $k-1$ are queueing. According to the above model, *any* of these $k-1$ nodes which are waiting in queue might leave the system independently with a rate γ. Any such departure will result in a move to state $k-1$. Such a move also occurs if the customer currently receiving service finishes (with rate μ). Thus, the correct CTMC for this queue is the following:

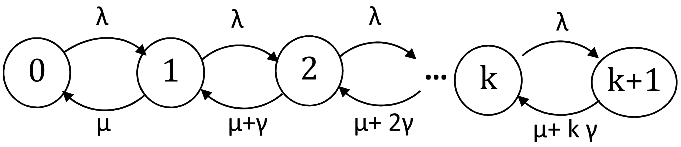

Figure 4: MC for Problem 5.1

Question 2: This is a birth-death chain, so we can use local balance equations:

$$\begin{aligned}
\pi_0\lambda &= \pi_1\mu \\
\pi_1\lambda &= \pi_2(\mu+\gamma) \\
\pi_2\lambda &= \pi_3(\mu+2\gamma) \\
\cdots & \\
\pi_k\lambda &= \pi_{k+1}(\mu+k\gamma)
\end{aligned}$$

Expanding this recursion gives

$$\pi_k = (\lambda^k)\pi_0 \prod_{i=0}^{k-1} \frac{1}{\mu+i\gamma}.$$

Finally, to get π_0: $\sum_{i=0}^{\infty}(\lambda^k)\pi_0 \prod_{i=0}^{k-1}\frac{1}{\mu+i\gamma} = 1$. That is

$$\pi_0 = \frac{1}{\sum_{k=0}^{\infty}(\lambda^k)\pi_0 \prod_{i=0}^{k-1}\frac{1}{\mu+i\gamma}}.$$

Question 3: This chain contains only a single class of states, as all states communicate with non-zero probability. Thus, they are either all transient or all recurrent. It is easy to see that, for whatever input rate λ, if we go far enough along the chain towards the right, there will be a large enough k such that $\lambda < \mu + k\gamma$. This means that this state will be recurrent (you could also formally prove this) and that the chain does not constantly grow to infinity.

Question 4: Customers are lost when they abandon the system. When at state k, the rate of lost customer is $(k-1)\gamma$. Averaging over all k,

$$\text{abandonment-rate} = \sum_{k=0}^{\infty} \pi_k (k-1)\gamma = (E[N]-1)\gamma$$

Question 5: Since this is a stable queue all customers that enter the system eventually receive service, except the ones that abandon the system. Thus, the total throughput X is

$$X = \lambda - \text{abandonment-rate} = \lambda - (E[N]-1)\gamma.$$

Problem 6 (extra credit - 20 points): A single cellular base station is used to cover users in a large area, as shown in Figure 5. When a user is in the "far" area, its service is exponential with rate q. When a user is in the "near" area, its service time is exponential with rate $1-q$. The operator can decide the parameter q by allocating the available frequencies to near and far users ($0 < q < 1$). It is further assumed that each user moves between the far and near areas according to the following Markov Chain (Note: assume for simplicity that a user will not change its area while it is being serviced).

- whenever in the "far" area a user moves to the "near" area with rate λ_2;
- whenever in the "near" area a user moves to the "far" area with rate λ_1;

- **Question 1 (8points) :** Assume there is only a single user in the cell, moving according to the above model. At some point, it generates a service request. What is the mean and the variance of the service time that it receives?
- **Question 2 (7 points) :** Assume there are now many users moving independently according to the above model. New service requests are generated in the cell with a (exponential) rate λ (this is the total load for the base station); Only 1 request can be served at a time (whether far or near), while the rest have to queue. What is the expected delay (queueing + service) for a new request?
- **Question 3 (5 points) :** How would you choose the value of q optimally (i.e. to minimize the expected delay of Question 2), as a function of λ_1 and λ_2?

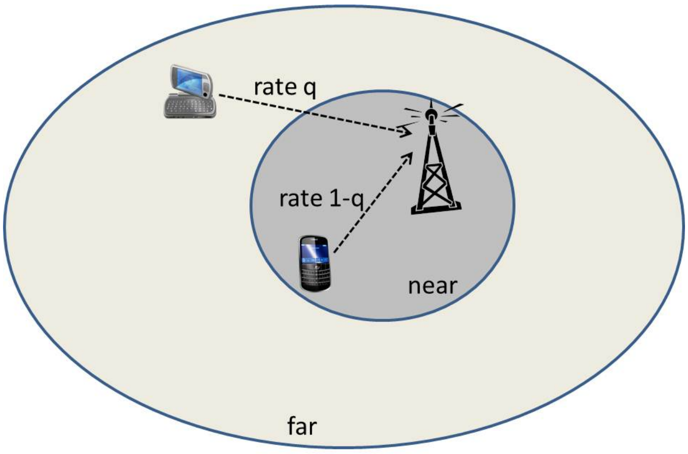

Figure 5: Problem 6

Solution:

Question 1: We just need to find the probability of the user being "far" or "near", when generating a service request. The user's mobility is subject to a simple 2-state Markov chain. Applying local balance to this chain we get:

$$\begin{aligned}\pi_{far}\lambda_2 &= \pi_{near}\lambda_1\\ \pi_{far}+\pi_{near} &= 1\end{aligned}$$

which give

$$\begin{aligned}\pi_{far} &= \frac{\lambda_1}{\lambda_1+\lambda_2}\\ \pi_{near} &= \frac{\lambda_2}{\lambda_1+\lambda_2}\end{aligned}$$

Thus, for the service time this user receives it holds that

$$\begin{aligned}E[S] &= \frac{\lambda_1}{\lambda_1+\lambda_2}\frac{1}{q}+\frac{\lambda_2}{\lambda_1+\lambda_2}\frac{1}{1-q}\\ Var[S] &= \frac{\lambda_1}{\lambda_1+\lambda_2}\frac{1}{q^2}+\frac{\lambda_2}{\lambda_1+\lambda_2}\frac{1}{(1-q)^2}\end{aligned}$$

Question 2: We can model this system as an M/G/1 queue, when there are many users competing for access to the single base station. Having derived the mean and variance for the service time of a single user (in question 1), we simply need to replace this in the pollaczek-khintchine formula to get the mean queueing delay $E[T_Q]$. As usual, the total response time is $E[T] = E[T_Q] + E[S]$.

Question 3: Based on questions 1 and 2, $E[T] = f(\lambda_1, \lambda_2, \lambda, q)$. We are asked to minimize $E[T]$ with respect to q, while keeping all other parameters constant. We need to solve

$$\frac{dE[T]}{dq} = 0.$$

We then pick the solution which lies between 0 and 1 (q must be a probability) and make sure that the second derivative is > 0 (to make sure it's a minimum, not a maximum). For those of you that have some background in convex optimization, you could include the constraint $(0 < q < 1)$ in the above equation using a Lagrange multiplier. The important thing for this last question was to understand how you should do it, not the actual numbers you get.