CSE 538 – Fall 2016 Midterm
4 Problems – 65 points total

Your Name:

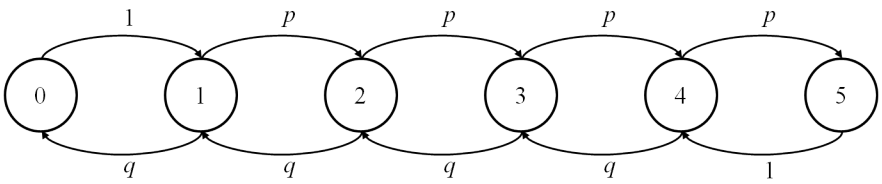

Figure 1: Discrete Time Markov Chain.

Problem 1 [10 points] Consider the Discrete Time Markov Chain (DTMC) of Fig. 1, where the parameters p and q satisfy $p + q = 1$.

1. **[5 points]** Does the chain admit a limiting distribution? If yes, does it for all combinations of p and q and why? If no, why not?

   The chain does not admit a limiting distribution as it is periodic for all values of p and q. If $pq \neq 0$, the period is 2, and if $pq = 0$, the period is also 2 but for different reasons. When $p = 1$ and $q = 0$, only states 4 and 5 have a non-zero (stationary) probability, which is $1/2$. Conversely, when $p = 0$ and $q = 1$, only states 0 and 1 have a non-zero (stationary) probability, which is again $1/2$.

2. **[5 points]** Can you find values for p and q such that $\pi_0 = \pi_1 = \pi_2 = \pi_3 = 0$ and $\pi_4 = \pi_5 = 1/2$. **Hint**: What does the balance equation for state 5 tells you?

   The balance equation for state 5 states that $p\pi_4 = \pi_5$, so that $\pi_4 = \pi_5$ is only feasible if $p = 1$ and, therefore, $q = 0$. Under theses assumptions, we readily get that $\pi_0 = \pi_1 = \pi_2 = \pi_3 = 0$, *i.e.,* the first four states are transient, and $\pi_4 = \pi_5 = 1/2$.

Problem 2 [25 points] Consider the closed system of Fig. 2 that consists of a dual CPU sub-system (CPU1+CPU2), where CPU1 can process 2 jobs per unit of time and CPU2 can process 4 jobs per unit of time, followed by a dual Disk sub-systems (Disk1+Disk2), where Disk1 can handle 1 job R/W operations per unit of time and Disk2 can handle 5 job R/W operations per unit of time. New jobs are assigned to CPU1 with probability p and to CPU2 with probability $1-p$. Similarly, when leaving the CPU sub-system, jobs are assigned to Disk1 with probability q and to Disk2 with probability $1-q$. The multi-programming level for the system is $N = 50$.

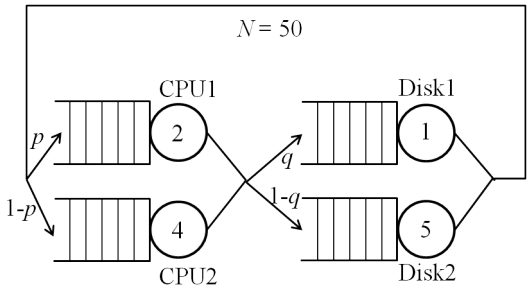

Figure 2: Closed system.

1. **[15 points]** Derive a tight upper bound for the number of jobs in the system, N^*, that corresponds to the threshold between low and high loads (multi-programming level or N value as per Theo. 7.1).
Hint: Consider the CPU and Disk sub-systems separately and determine their individual thresholds.

   In order to characterize N^*, we first characterize performance for the CPU and Disk sub-systems separately. Specifically, we have

$$\begin{aligned} E[V_{\text{CPU1}}] &= p, E[V_{\text{CPU2}}] = 1-p, \quad E[S_{\text{CPU1}}] = \frac{1}{2}, E[S_{\text{CPU2}}] = \frac{1}{4} \\ E[V_{\text{Disk1}}] &= q, E[V_{\text{Disk2}}] = 1-q, \quad E[S_{\text{Disk1}}] = 1, E[S_{\text{Disk2}}] = \frac{1}{5} \end{aligned}$$

   This implies

$$\begin{aligned} E[D_{\text{CPU1}}] &= \frac{p}{2}, \quad E[D_{\text{CPU2}}] = \frac{1-p}{4} \\ E[D_{\text{Disk1}}] &= q, \quad E[D_{\text{Disk2}}] = \frac{1-q}{5}, \end{aligned}$$

   and therefore, we have

$$D_{\text{CPU}} = \frac{p}{2} + \frac{1-p}{4} = \frac{p+1}{4}, \quad \text{and} \quad D_{\max}^{\text{CPU}} = \max\left(\frac{p}{2}, \frac{1-p}{4}\right),$$

   and

$$D_{\text{Disk}} = q + \frac{1-q}{5} = \frac{4q+1}{5} \quad \text{and} \quad D_{\max}^{\text{Disk}} = \max\left(q, \frac{1-q}{5}\right).$$

   Focusing on the CPU sub-system first, we have

$$N_{\text{CPU}}^* = \frac{D_{\text{CPU}}}{D_{\max}^{\text{CPU}}} = \frac{\frac{p+1}{4}}{\max\left(\frac{p}{2}, \frac{1-p}{4}\right)} = \begin{cases} \frac{\frac{p+1}{4}}{\frac{1-p}{4}} = \frac{p+1}{1-p} \le \frac{\frac{4}{3}}{\frac{2}{3}} = 2 & p \le \frac{1}{3} \\ \frac{\frac{p+1}{4}}{\frac{p}{2}} = \frac{p+1}{2p} = \frac{1}{2} + \frac{1}{2p} \le \frac{1}{2} + \frac{3}{2} = 2 & p \ge \frac{1}{3} \end{cases}$$

   So in all cases $N_{\text{CPU}}^* \le 2$.

Following a similar approach for the Disk sub-system, we get

$$N^*_{\text{Disk}} = \frac{D_{\text{Disk}}}{D^{\text{Disk}}_{\max}} = \frac{\frac{4q+1}{5}}{\max\left(q, \frac{1-q}{5}\right)} = \begin{cases} \frac{\frac{4q+1}{5}}{\frac{1-q}{5}} = \frac{4q+1}{1-q} \leq \frac{\frac{4}{6}+1}{\frac{5}{6}} = \frac{10}{5} = 2 & q \leq \frac{1}{6} \\ \frac{\frac{4q+1}{5}}{q} = \frac{4}{5} + \frac{1}{5q} \leq \frac{4}{5} + \frac{6}{5} = 2 & q \geq \frac{1}{6} \end{cases}$$

So that again in all cases $N^*_{\text{Disk}} \leq 2$.

Hence, $N^* \leq 4 \ll 50$. This is because $D = D_{\text{CPU}} + D_{\text{Disk}}$ and $D_{\max} = \max\left(D^{\text{CPU}}_{\max}, D^{\text{Disk}}_{\max}\right)$, and

$$\frac{A+B}{\max(a_1, a_2, b_1, b_2)} \leq \frac{A}{\max(a_1, a_2)} + \frac{B}{\max(b_1, b_2)}$$

Numerical computations readily give a tighter bound of about 2.8.

2. **[5 points]** What values of p and q minimize the system response time $E[R]$? Provide an explicit upper bound for $E[R]$.

   The system response time is minimized by separately minimizing the response times of the CPU and Disk sub-systems.
   From the previous question, we know that for the CPU sub-system $N^*_{\text{CPU}} \leq 2 \ll 50$, so that minimizing the response time of the CPU sub-system calls for minimizing $D^{\text{CPU}}_{\max}$. This can be readily seen to be realized through load-balancing, *i.e.,* set $p = \frac{1}{3}$, which yields $D^{\text{CPU}}_{\max} = \frac{1}{6}$.
   A similar reasoning applies to the Disk sub-system for which $N^*_{\text{Disk}} \leq 2 \ll 50$, so that minimizing the response time of the Disk sub-system also calls for minimizing $D^{\text{Disk}}_{\max}$. This is again realized through load-balancing, *i.e.,* set $q = \frac{1}{6}$, which yields $D^{\text{Disk}}_{\max} = \frac{1}{6}$.
   This in turn implies that the $D_{\max}$ value for the entire system is also $D_{\max} = \frac{1}{6}$ so that the response time $E[R]$ is upper-bounded by $E[R] \leq \frac{50}{6} = \frac{25}{3} = 8.333$.

3. **[5 points]** You are now considering either replacing the two CPUs with a single faster CPU of speed 10 or alternatively the two disks with a single faster (and bigger) disk also of speed 10. Does either of these two options meaningfully improve the system response time. Justify your answer.

   The two sub-systems have the same $D_{\max}$ value so that improving the performance of one won't improve the performance of the other, and since $N \gg N^*$, the performance of the overall system won't be affected either. Hence, neither of the two proposed replacements yields meaningful improvements in performance.

Problem 3 [20 points] Consider a system where jobs arrive according to a Poisson process of rate λ, and have service times whose duration is exponentially distributed with mean $1/\mu$. Jobs are, however, impatient, and, as illustrated in Fig. 3, each job that waits in the queue leaves after an exponentially distributed time also of mean $1/\mu$. In other words, jobs can leave the queue before they reach the server.

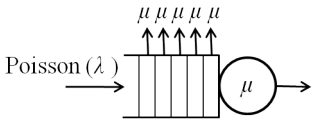

Figure 3: System with "impatient" customers.

1. **[5 points]** Give a Markov chain representation for the system, where the state is the number of jobs.

   Given that there are $i \geq 0$ customers in the system, transitions to state $i+1$ occur with rate λ, the arrival

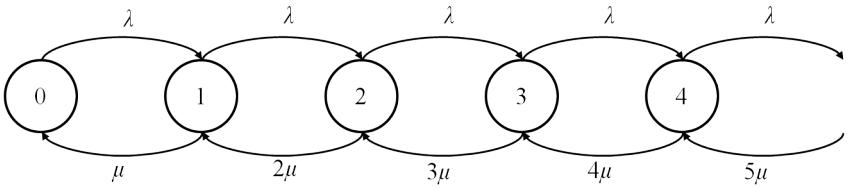

Figure 4: Markov chain for system with "impatient" customers.

   rate. Conversely, transitions from state $i > 0$ to state $i-1$ occur with rate $\mu + (i-1)\mu = i\mu$. This due to the fact that the job in service leaves with a rate μ, while the $(i-1)$ jobs in the queue each leave also with rate μ. The resulting Markov chain is shown in Fig. 4. Note that the chain is essentially identical to that of the M/M/∞ queue, which is not surprising since whether in service or waiting in the queue, jobs leave at the same rate of μ, *i.e.,* a waiting spot in the queue is indistinguishable from the server when it comes to its effect on state transitions.

2. **[10 points]** Assuming that the chain is ergodic, provide an expression, function of λ and μ, for the probability π_i that there are i jobs in the system.

   The balance equations can be written as

$$\pi_i = \frac{\rho^i}{i!}\pi_0, \quad i \geq 0$$

   where $\rho = \frac{\lambda}{\mu}$, so that the normalization equation is of the form

$$\pi_0 = \left[\sum_{i=0}^{\infty} \frac{\rho^i}{i!}\right]^{-1} = e^{-\rho}$$

   which is positive for all finite values of ρ.

3. **[5 points]** Based on the expression of π_i, propose a *simple* condition that ensure that the chain is ergodic.

   As stated above we have $\pi_0 = e^{-\rho} > 0\,, \forall \rho < \infty$. Hence, the chain is ergodic for all values of λ as long as $\mu > 0$.

Problem 4 [10 points] Consider a single server queueing system with an infinite waiting room, and two types of jobs, where jobs of type $i, i = 1, 2$, arrive according to a Poisson process of rate λ_i. The service times of both types of jobs are exponentially distributed with mean $1/\mu$, where $\mu > \lambda_1 + \lambda_2$. What is the expected number of type i jobs in the system as a function of λ_1, λ_2 and μ?

The system behaves as an M/M/1 queue with total arrival rate $\lambda = \lambda_1 + \lambda_2$ and service rate μ. From the solution of the M/M/1 queue, we know that the expected time in the system of a random job is

$$E[T] = \frac{1}{\mu - \lambda}$$

From PASTA, we know that both types of jobs sample the system at random times, and therefore also experience an average system time of $E[T]$. Applying Little's Law, we then get

$$E[N_i] = \frac{\lambda_i}{\mu - \lambda}$$