CSE 538 – Fall 2014 Midterm Quiz

6 questions – 120 points total – 100 points max

Questions flagged with a “ * ” are harder

Your Name:

1. **[10 points]** Consider a general closed network representing a batch system with a *fixed* number of jobs N circulating through the system. Internally, the system includes various "design parameters," *e.g.,* load balancing options that can be adjusted to affect its performance. Is it possible for a system designer to select those parameters to *simultaneously* maximize throughput and minimize the average response time. Rigorously justify your answer.

   Little's Law for a closed batch system states that

   $$N = X \cdot E[R]\,,$$

   where X is the system throughput and $E[R]$ its average response time. Assuming that N is fixed, this readily implies that maximizing X is equivalent to minimizing $E[R]$, *i.e.,* the design that realizes the maximum throughput also minimizes the system's response time.

2. **[25 points]** Consider the closed network of Fig. 1 that represents the operation of a multi-processor system

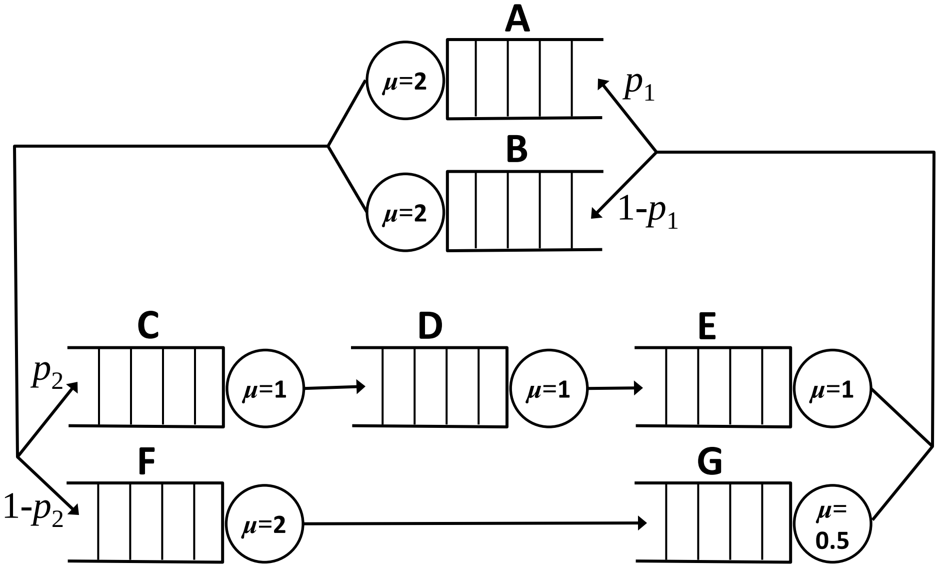

Figure 1: Closed network model of multi-processor system.

   consisting of seven (7) CPUs labeled A to G. The average job processing rate μ of each CPU is as indicated on the figure with CPUs A, B, and F capable of processing 2 jobs/sec, CPUs C, D, and E capable of processing 1 job/sec, and CPU G taking 2 secs on average to process one job.

   The system also include two *load-balancers.* As shown in Fig. 1, one assigns jobs to CPUs A and B in the ratio p_1 and $1-p_1$, $0 \leq p_1 \leq 1$, respectively, and the second performs a similar task for CPUs C and F with allocation ratios p_2 and $1-p_2$, $0 \leq p_2 \leq 1$, respectively.

   Consider separately the cases $N = 1$ and $N = 10$, *i.e.,* a system with only one job circulating versus one with 10 jobs, and for each case identify the values of p_1 and p_2 that maximize the system throughput X. **Hint**: Identify first how X depends on p_1 and p_2 for a general N value, and then determine which part of the expression is relevant for $N = 1$ and $N = 10$.

   Let $D_i, i = A, B, C, D, E, F, G$, denote the total service demand on each one of the seven CPUs for all visits of a single job. We have the following expressions for $E[D_i]$:

$$\begin{aligned} E[D_A] &= \frac{p_1}{2}, E[D_B] = \frac{(1-p_1)}{2}, \\ E[D_C] &= p_2, E[D_D] = p_2, E[D_E] = p_2 \\ E[D_F] &= \frac{(1-p_2)}{2}, E[D_G] = 2(1-p_2) \end{aligned}$$

This implies that $D = \sum_{i=A}^{G} E[D_i]$ is of the form

$$D = \frac{p_1}{2} + \frac{(1-p_1)}{2} + 3p_2 + (1-p_2)\frac{5}{2} \quad \Rightarrow D = 3 + \frac{p_2}{2}\,.$$

Similarly, we also have

$$D_{\max} = \max\left\{\frac{p_1}{2}, \frac{(1-p_1)}{2}, p_2, 2(1-p_2)\right\}$$

Since the throughput X of the system is upper-bounded by

$$X \leq \min\left(\frac{N}{D}, \frac{1}{D_{\max}}\right)$$

maximizing X calls for minimizing either D or $D_{\max}$. Which of the two is the limiting factor depends on the value of N, *i.e.,* for $N \leq N^* = \frac{D}{D_{\max}}$, D is the relevant term, while $D_{\max}$ is when $N \geq N^*$.

From the above expression for D, we immediately obtain that

$$3 \leq D \leq 3.5\,.$$

The lower bound is obtained for $p_2 = 0$ and the upper bound for $p_2 = 1$. Similarly, when considering $D_{\max}$, we know that

$$D_{\max} \geq \max\left\{\frac{1}{4}, \frac{2}{3}\right\} = \frac{2}{3}\,,$$

where we have used the fact that

$$\frac{1}{4} = \min_{0\leq p_1 \leq 1}\left\{\frac{p_1}{2}, \frac{(1-p_1)}{2}\right\} \quad \text{and} \quad \frac{2}{3} = \min_{0\leq p_2\leq 1}\{p_2, 2(1-p_2)\}\,,$$

where the first minimum is realized for $p_1 = \frac{1}{2}$ and the second minimum is realized for $p_2 = \frac{2}{3}$. We also trivially have that $D_{\max} \leq 2$ (this upper bound is achieved when $p_2 = 0$). From these inequalities, we conclude that $1.5 \leq N^* \leq 5.25$. Hence, when $N = 1$, we need to minimize D, and when $N = 10$, we need to minimize $D_{\max}$.

As mentioned above, minimizing D calls for setting $p_2 = 0$, *i.e.,* never use the higher average latency "branch" ($1+1+1 = 3$ through C-D-E versus only $0.5 + 2 = 2.5$ through F-G). Note that p_1 can take any value, since with only 1 job in the system, there is no benefit to having parallel processing branches A and B. Hence, when $N = 1$, the throughput is maximized by setting $p_2 = 0$ and is then equal to $X_1 = \frac{1}{3}$.

In contrast, when $N = 10 > N^*$ so that maximizing throughput calls for minimizing $D_{\max}$, we need to set $p_2 = \frac{2}{3}$ while p_1 can again take any value. The latter is that either CPU A or CPU B alone is "faster" than the combination of the two lower branches that has an aggregate throughput of 1.5 (1 for the top branch and 0.5 for the lower branch). Setting $p_2 = \frac{2}{3}$ means that we send twice as many jobs to the top branch that has twice the throughput of the lower branch, and hence realize a throughput $X_{10} \approx \frac{3}{2} = 1.5$. Note that this value is actually only an upper bound that is realized only for N large enough (it requires that all the CPUs of the two lower branches always be busy), but we should be reasonably close to realizing this when $N = 10$.

3. **[15 points]** Consider a single processor computer system that operates according to a discrete-time clock. The maximum number of jobs that can be accommodated by the system is limited to 64. The measured average inter-arrival time is I clock cycles, the measured average service time is S clock cycles, and measurements also tell us that the processor is busy 83% of the time. Based on these measurements, what is the fraction of arriving jobs that are discarded?
   Hint: Think how Little's Law can help relate the different quantities of interest, and the fact that discarded jobs don't make it to the processor.

   Little's Law states that the expected number of jobs in the processor (or alternatively, the fraction of time the processor is busy) is equal to the arrival rate A' to the processor times the average service time S of a job. In other words,

$$0.83 = A'S \quad \Rightarrow \quad A' = \frac{0.83}{S}\,.$$

   Since measurements tell us that we get on average $A = \frac{1}{I}$ jobs/sec, the fraction d of jobs that are discarded is given by

$$d = \frac{A - A'}{A} = 1 - \frac{0.83 \times I}{S}$$

   Note that is we make further assumptions on when arrivals and departures take place, *e.g.,* arrivals at the start of a clock cycle and departures at the end, then we can compute explicit state probabilities, including $\pi_{64} = P_b = \frac{1-\rho}{1-\rho^{65}}\rho^{64} \approx 1.13 \times 10^{-6}$, where $\rho = \frac{S}{I}$. The value of ρ can be computed based on the fact that $\pi_0 = 1 - 0.83 = 0.17 = \frac{1-\rho}{1-\rho^{65}}$.

4. **[20 points]** Consider the DTMC shown in Fig. 2 that has a state dependent arrival rate. Specifically, when

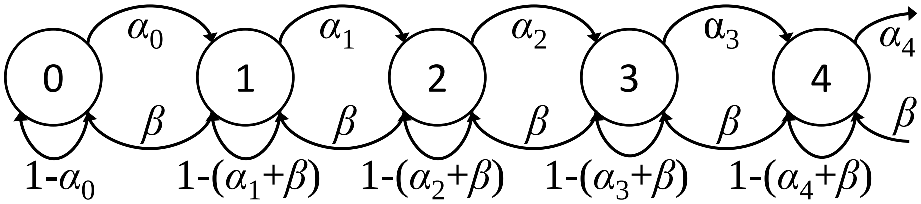

Figure 2: Discrete-time Markov chain (DTMC).

   the chain is in state $i, i \geq 0$, the probability of an arrival and no departure while in state i is equal to $\alpha_i = \frac{\alpha}{i+1}$. This is an instance of so-called *discouraged* arrivals, *i.e.,* high system occupancy (as measured by i) discourages new arrivals. Conversely, the probability of no arrival and a departure while in state $i, i \geq 0$, is constant and equal to β.

   Under these assumptions find an expression for π_0, function of α and β, and use it to identify under what conditions, again as a function of α and β, a stationary distribution exists for the chain.
   Hint: In deriving an expression for π_0, remember the Poisson distribution.

   From Fig. 2, we readily see that the balance equations for the chain are of the form

$$\alpha_i\pi_i = \beta\pi_{i+1}, i \geq 0\,, \quad \Rightarrow \pi_{i+1} = \frac{\alpha}{\beta}\frac{1}{i+1}\pi_i, i \geq 0\,, \quad \Rightarrow \pi_i = \frac{\rho^i}{i!}\pi_0, i \geq 0\,, \text{ where } \rho = \frac{\alpha}{\beta}\,.$$

   The normalization condition $\sum_{i=0}^{\infty} \pi_i = 1$ implies that π_0 satisfies

$$\pi_0 \sum_{i=0} \frac{\rho^i}{i!} = 1 \quad \Rightarrow \pi_0 = e^{-\rho}\,,$$

where the last equality comes from the Poisson distribution that states that $\sum_{i=0}^{\infty} \frac{\rho^i}{i!} e^{-\rho} = 1$. Note that the above expression always satisfies $0 < \pi_0 \leq 1$ as long as $\rho \geq 0$, which is always verified for all values of $\alpha, \beta \geq 0$. Hence, the chain always admits a stationary distribution.

5. **[20 points]** Consider a dual-core processor with 2 identical cores that operate with the same clock. Jobs arrive to the processor system according to an arrival process that has a geometrically distributed inter-arrival time with average duration $\frac{1}{p}$, *i.e.,* p is the probability that there is an arrival in a given slot. Jobs have service times that are also geometrically distributed with a mean duration of $\frac{1}{q}$ clock cycles. Job arrivals are at the beginning of a clock cycle, while job departures take place at the end of a clock cycle, *i.e.,* a single cycle job can arrive and depart in the same clock cycle. Jobs that arrive to the processor and find both cores busy are dropped (or redirected to another processor). Jobs that arrive to an idle processor (both cores are idle) are assigned randomly to either core.

   Formulate a Markov chain representation for the system and identify its transition probability matrix P as a function of p and q. Use P to compute the probability P_b that a job is dropped, *i.e.,* the fraction of jobs that are discarded, for the special case $p = q = \frac{1}{2}$.

   Let i denote the number of cores that are busy at the end of a clock cycle. Both arrivals and departures are driven by geometric distributions, so that the system state as defined readily verifies the Markov property. The state transition probabilities matrix P for the system verifies

$$P = \begin{bmatrix} (1-p)+pq & p(1-q) & 0 \\ q(1-p)+q^2p & 2qp(1-q)+(1-q)(1-p) & p(1-q)^2 \\ q^2 & \binom{2}{1}q(1-q) & (1-q)^2 \end{bmatrix},$$

   where we have accounted for the fact that jobs can arrive and depart in the same clock cycle.

   Jobs are blocked if they arrive to see both cores busy, which occurs with probability π_2, *i.e.,* $P_b = \pi_2$. In order to compute π_2, we need to obtain the solution of $\pi = \pi P$ for $p = q = \frac{1}{2}$. The corresponding system of linear equations is of the form

$$\begin{aligned} \pi_0 &= \frac{3}{4}\pi_0 + \frac{3}{8}\pi_1 + \frac{1}{4}\pi_2 \\ \pi_1 &= \frac{1}{4}\pi_0 + \frac{1}{2}\pi_1 + \frac{1}{2}\pi_2 \\ \pi_2 &= \frac{1}{8}\pi_1 + \frac{1}{4}\pi_2 \end{aligned}$$

   which together with the normalization equation $\pi_0+\pi_1+\pi_2 = 1$ readily gives $\pi_0 = \frac{10}{17}, \pi_1 = \frac{6}{17}, \pi_2 = \frac{1}{17}$. Hence, we have $P_b = \frac{1}{17}$.

6. **[$10 + 20^* = 30$ points]** Consider a single processor computer system that operates according to a discrete-time clock. The system is capable of accommodating an “infinite” number of jobs waiting to access the processor. In each time-slot, a job arrives a the beginning of a slot with probability $p, 0 < p < 1$. Jobs are of two types, high and low priority. Upon arrival of a job, it is identified as high priority with probability $\alpha, 0 < \alpha < 1$, and low priority with probability $1-\alpha$, so that αp is the arrival rate of high priority jobs and $(1-\alpha)p$ the arrival rate of low priority jobs. All jobs (low and high priority) have a processing time that is geometrically distributed with a mean duration of $\frac{1}{q}$ clock cycles, *i.e.,* when in service a job leaves at the end of a slot with probability q. Note that a job can arrive (at the beginning) and depart (at the end) in the same time-slot.

   When a high priority job arrives to the system, it is either enqueued in first-in-firt-out (FIFO) order behind any other high priority jobs waiting, or immediately moves into service, possibly preempting any low priority job that may be in service at the time. Low priority jobs are served in FIFO order only when there are no high priority jobs in the system. A preempted low-priority job resumes its service where it had left it when preempted, *i.e.,* proceeds to complete its residual service.

(a) **[10 points]** Obtain an expression, function of the arrival and service parameters, for the probability π_0 that the system is empty.
Hint: The priority structure and preemption of low priority jobs does not really affect how the system processes work, and therefore the odds that the server be idle.

Because preempted low priority jobs resume their service where they had left it, their service time is not affected, so from the point of view of the system operation, the fact that a low priority job may be swapped out of service does not affect the number of jobs in the system, when they arrive, and when departure takes place (because of the geometric distribution of the service time). Hence, this system has the same balance equations as a system without priority, *i.e.,* $\pi_i = \rho^i \pi_0$, where $\rho = \frac{p(1-q)}{q(1-p)}$ and $\pi_0 = 1 - \rho$.

As a side comment, note that for this system $1 - \pi_0$ is *not* the probability that the server is busy. This is because we sample the system at the end of a slot, and when the system is idle at the start of a slot but there is arrival and a departure in the same slot, the server will be busy (in that slot), but we do not count that slot in the system statistics. Instead, the probability or fraction of time that the server is busy is given by

$$\begin{aligned} P(\text{server busy}) &= 1 \times \sum_{i=1}^{\infty} \pi_i + p\pi_0 = 1 - \pi_0 + p\pi_0 \\ &= \rho + p(1-\rho) = \frac{p(1-q)}{q(1-p)} + \left(1 - \frac{p(1-q)}{q(1-p)}\right) p \\ &= \frac{p(1-q) + (q-p)p}{q(1-p)} = \frac{p - pq + pq - p^2}{q(1-p)} \\ &= \frac{p}{q}, \end{aligned}$$

where we note that the latter is consistent with what we expect from Little's Law, since the arrival rate to the system is p and the average time a job spends in service is $\frac{1}{q}$.

(b) **[20 points]*** Obtain an expression, function of the arrival and service parameters, for the average time in the system, $E[T_L]$, for low priority jobs.
Hint 1: The low priority jobs are essentially transparent to the high priority ones.
Hint 2: Because of the memoryless property of the geometric distribution, the residual service time of a preempted (low priority) job is indistinguishable from that of the service time of the high priority job that preempts it and moves into service, *i.e.,* it (statistically) makes no difference to the overall system, which one is served first.
Hint 3: Remember that Little's Law can be applied to any "logical" part of a system. Consider applying it twice, to two different logical pieces of the system.

Since the low priority jobs are essentially transparent to the high priority ones, we can obtain the stationary probabilities $\pi_i^H, i \geq 0$, that there are i high priority customers in the system simply by considering a single processor system with geometric arrivals with probability $p_H = \alpha p$, and geometrically distributed service times with mean service time $\frac{1}{q}$. Given that jobs can arrive and depart in the same slot, we readily find that $\pi_i^H = \rho_H^i(1 - \rho_H)$, where $\rho_H = \frac{\alpha p(1-q)}{q(1-p)}$. This then allows to compute the average number of high priority jobs in the system $E[N_H]$ as follows

$$E[N_H] = \sum_{i=0}^{\infty} \left[i\pi_i^H (1-pq)\alpha pq + (i+1)\pi_i pq \right] ,$$

where the two terms account for the fact that when in state i during a clock cycle, there can either be i jobs (whenever there are no arrivals and departures during that slot, which has probability $1 - \alpha pq$) or $(i+1)$ jobs (whenever one job arrives at the beginning of the slot and one leaves at the end, which has probability αpq). This then gives

$$E[N_H] = \alpha pq + \frac{\rho_H}{1-\rho_H}$$

Similarly, the total (high and low priority) average number of jobs in the system $E[N]$ is given by

$$E[N] = pq + \frac{\rho}{1-\rho} \qquad \text{where as before } \rho = \frac{p(1-q)}{q(1-p)}$$

Since $N = N_H + N_L$, we know that $E[N] = E[N_H] + E[N_L]$, so that $E[N_L] = E[N] - E[N_H]$, which gives

$$E[N_L] = (1-\alpha)pq + \frac{\rho_L}{(1-\rho)(1-\rho_H)} ,$$

where $\rho_L = \frac{(1-\alpha)p(1-q)}{q(1-p)}$. Applying Little's Law to the "system" consisting only of low priority jobs, we have that $E[N_L] = (1-\alpha)pE[T_L]$, which then gives

$$E[T_L] = q + \frac{q(1-q)(1-p)}{(q-p)(q(1-p) - \alpha p(1-q))}$$