Performance Evaluation of Computer Systems and Networks, 15/06/2021

Exercise 1

A *schedule* is a sequence of *events*, which can be of two types, *A* and *B*:

- The probability of an event of type *A* is 0.4, and the one of an event *B* is 0.6;
- Event *A* lasts for 4ms, and event *B* lasts for 5 ms.
- Events are independent of each other.

Let S_n denote the RV that measures the time duration of a schedule of n events.

1) Find the probability mass function of S_4 and its mean value $E[S_4]$.
2) Compute the PMF of S_n and $E[S_n]$, for a generic value of n.

3) Assume now that the distribution of the *number* of events in a schedule is as follows: there are never less than five events, and the probability of having more than five events is $p_{5+n} = \left(\frac{1}{2}\right)^{n+1}$, with $n \geq 0$. Under the above hypothesis, you measure 40ms as the duration of the schedule. Compute the PMF of the number of events in that schedule.

Exercise 2

A multiprogrammed computer system has N running processes. These processes request I/O operations independently, each at a rate λ, with exponentially distributed inter-request times. I/O operations are served by an array of N identical I/O peripherals. Each I/O operation has a duration which is exponential with a mean $\frac{1}{\mu}$.

1) Model the system as a queueing system and draw the transition rate diagram
2) Compute the steady-state probabilities and the stability condition
3) Compute the mean response time for a process
4) Compute the mean number of processes blocked on an I/O operation
5) Compute the throughput of the I/O subsystem. Interpret the result.

Exercise 1 - Solution

1) A sequence of four events can last for 16, 17, 18, 19, 20 ms, depending on the number of Bs in it. If you consider each digit as a Bernoullian experiment, the probability of having k Bs is $\sim B(4{,}0.6)$, hence we have:

$P\{S_4 = 16 + k\} = \binom{4}{k} 0.6^k \cdot 0.4^{4-k}$, with $0 \le k \le 4$.

The mean value for S_4 is $16 + 4 \cdot 0.6 = 18.4$.

2) in the general case, S_n is an integer ranging from $4n$ to $5n$, and we have:

$P\{S_n = 4 \cdot n + k\} = \binom{n}{k} 0.6^k \cdot 0.4^{n-k}$, with $0 \le k \le n$.

The mean value for S_n is $(4 + 0.6) \cdot n = 4.6 \cdot n$.

3) We first need to compute all the values of n for which $4n \le 40 \le 5n$. They are the intersection of $n \le 10$ (left-hand inequality) with $n \ge 8$ (right-hand inequality). Hence, the transmission can be of 8, 9 or 10 events. The "*a priori*" probabilities of a transmission of 8, 9, 10 digits are $\frac{1}{16}, \frac{1}{32}, \frac{1}{64}$, respectively. After you observe that the transmission is 40ms long, you can apply Bayes's theorem and compute the "*a posteriori*" probabilities as follows:

$$P(n|40) = \frac{P(40|n) \cdot P_n}{\sum_{k=8}^{10} P(40|k) \cdot P_k},$$

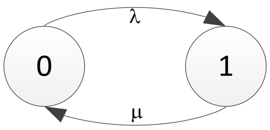
with $n = 8{,}9{,}10$.

The *a posteriori* probabilities can be found in the rightmost column of the following table:

n	P(40|n)	P(40|n)	Pn	Pn	product	P(n|40)
8	$\binom{8}{8} 0.6^8$	0.016796	$\frac{1}{16}$	0.0625	0.00105	16.72%
9	$\binom{9}{4} 0.6^4 \cdot 0.4^5$	0.167215	$\frac{1}{32}$	0.03125	0.005225	83.25%
10	$\binom{10}{0} 0.4^{10}$	0.000105	$\frac{1}{64}$	0.015625	1.64E-06	0.03%
				total	0.006277	

Exercise 2 - Solution

There are two ways to model this system. The simplest is to observe that, since there are as many I/O peripherals as running processes, no queueing ever occurs, hence the system can be seen as the juxtaposition of N independent M/M/1/1 systems as the one in the figure:



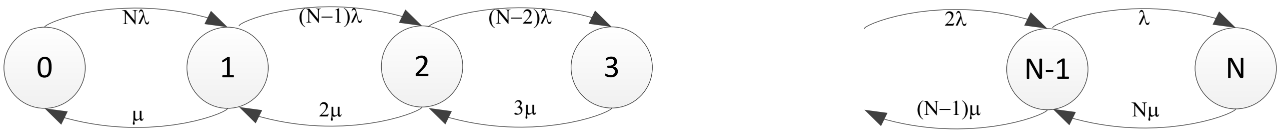
State "1" is the one where the process is doing some I/O operation, state "0" is when it is not occupying the I/O peripheral. The above system admits two SS probabilities, $\pi_0 = \frac{1}{(1+u)}$, $\pi_1 = \frac{u}{(1+u)}$, with $u = \frac{\lambda}{\mu}$, and is always stable. Therefore, we can define the state of the multiprogrammed system as the number of occupied I/O peripherals, ranging from 0 to N.

The probability that k peripherals are occupied is thus a binomial RV, with a number of trials N and a probability of success equal to π_1, hence:

$P_k = \binom{N}{k} {\pi_1}^k \cdot (1-\pi_1)^{N-k} = \binom{N}{k} {\pi_1}^k \cdot {\pi_0}^{N-k} = \binom{N}{k} \cdot \frac{u^k}{(1+u)^N}.$

The system has a finite number of states, hence it is always stable.

The same solution could be found using the canonical procedure, i.e. seeing the system as an M/M/N/./N one as follows:



Developing the computations yields the selfsame results.

3) The mean response time for a process is $\frac{1}{\mu}$, since there is no queueing.

4) The mean number of blocked processes is the mean of the binomial, i.e., $N \cdot \pi_1 = \frac{N \cdot u}{(1+u)}$.

5) The throughput of the I/O subsystem is

$$\gamma = \sum_{n=1}^{N} \mu_n \cdot P_n = \sum_{n=1}^{N} n \cdot \mu \cdot \binom{N}{n} \cdot \frac{u^n}{(1+u)^N} =$$

$$= \sum_{n=1}^{N} \lambda \cdot \frac{N!}{(n-1)! \cdot (N-n)!} \cdot \frac{u^{n-1}}{(1+u)^N} =$$

$$= N \cdot \frac{\lambda}{1+u} \cdot \sum_{n=1}^{N} \frac{(N-1)!}{(n-1)! \cdot \big(N-1-(n-1)\big)!} \cdot \frac{u^{n-1}}{(1+u)^{N-1}} =$$

$$= N \cdot \frac{\lambda}{1+u} \cdot \sum_{n=0}^{N-1} \binom{N-1}{n} \cdot \left(\frac{u}{1+u}\right)^n \cdot \left(\frac{1}{1+u}\right)^{(N-1)-n} =$$

$$= N \cdot \frac{\lambda}{1+u} = N \cdot \mu \cdot \frac{u}{1+u}$$

The last result is easily explained, since it states that the throughput is equal to the serving rate of each I/O peripheral, times the probability that the latter is occupied, times their overall number N.