CEN 591, Fall 2012
Performance Lab: Performance evaluation using simulation and analytical models
Assigned: Sept. 24, Due: Sun, Oct. 7, 11:59PM

1 Introduction

The purpose of this assignment is to understand the simulation and analytical based performance evaluation of computer systems using M/M/1 and M/M/s queuing model.

2 Logistics

This assignment must be done individually. All handins are electronic. Clarifications and corrections will be posted on the blackboard

3 Handout Instructions

You can download the assignment handout, `HW2-handout.tar`, from the course Web page.

Start by copying `HW2-handout.tar` to a directory on a Linux machine in which you plan to do your work. Then give the command

```
unix> tar xvf HW2-handout.tar
```

This will cause SMPL related files to be unpacked that you will need them to develop the required programs. SMPL is an event based simulator that is originally written by M. H. MacDougall and described in *Simulating Computer Systems, Techniques and Tools, The MIT Press, 1987.* To install SMPL refer to the README in the main directory. Also refer to `doc` directory to see how to use SMPL. In this assignment you need to use/modify `mm1q.c`. In `mm1q.c` the variable `Ta` denotes the average inter-arrival time of requests, `Ts` denotes the average service time, and `Te` denotes the simulation time.

4 Preliminaries

In many computer systems, a job has to pass through a series of processes arranged in a network structure. Consider each process as a queue. In a special case where all arrivals to each process in the network follows a Poisson process, all service times are exponentially distributed, there is no limit on the processes' buffer (infinite number of tasks can wait for the processes), and task' flow in the network is independent of their past histories, the network is called open Jackson network without feedback. Such a network is a collection of connected M/M/1 (or M/M/s) queues with known parameters. In this case each process can be analyzed separately using M/M/1 or M/M/s model. Mean delays for each process can be added to determine mean system (network) delay.

In this assignment you are asked to develop a Jackson network (open network with no feedback) using SMPL simulator to simulate the given computer system performance. You should also evaluate the system performance analytically. Finally you'll compare the results.

Merging and multiplexing of workload in M/M/1: (i) in M/M/1 as long as the arrival rate is less than the service rate (i.e. $\lambda < \mu$), the release rate is also a Poisson process with the rate λ, (ii) the merge of two Poisson processes with the rate λ_1 and λ_2 is a Poisson process with the rate of $\lambda_1 + \lambda_2$.

5 Question 1 (50 points)

Consider an Internet application in a data center that is composed of three tiers. Users' request must first pass through the first tier (e.g., web tier). 90% of requests goes through and 10% are rejected. All accepted requests will pass through the second tier (e.g., application tier). However, only 60% are routed to the third tier (e.g., database tier).

Requests arrive randomly , such that their inter-arrival time follows exponential distribution with the average of 5 ms.

The average service times of all the tiers are exponentially distributed and have the mean time of 3, 5, and 5 ms, respectively.

5.1 Submission format

- Write a C program to simulate the given network (hint: use `mm1q.c` example of SMPL), develop a Makefile to automate your program compilation and a bash script named `perfeval.sh` to produce the results. `mm1q` outputs that can be used are as follows: `Util`, i.e., the average utilization, `MEAN BUSY PERIOD`, i.e, the average service time, and `MEAN QUEUE LENGTH`, i.e., the average queue length. Carefully adjust the simulation parameters including the simulation time. Your program should prints 24 lines. The first seven lines should be the parameter values of the first tier M/M/1 queue as follows: mean arrival rate, mean service time, mean utilization, mean queue waiting time , mean queue length, mean waiting time in the system, and mean number of request in the system. The next following 14 lines should be the values of the second and thirds tier M/M/1 queues parameters in the same order as the first tier's. The last line should be the mean waiting time of a request in the network.

- Calculate all of the aforementioned 24 parameters analytically. Create a text file and name it as `analytical-result` and print your analytical results in the file in the same order that your C program prints the outputs.
- Report your analytical results in a document, and answer the following questions in the document.
  - Do you observe any difference in analytical and simulation results? why? if you observe differences, can you suggest a solution to decrease the differences?
  - What changes do you need to to do such that the average delay of requests becomes less than 20 ms. Analytically show the average delay.

6 Question 2 (25 points)

Request inter-arrival time for a system is 55 ms. Currently the system has only one server which processes the requests with the average service time of 50ms. Both the inter-arrival time and service time follow exponential distribution. We want to improve the system such that the average delay (waiting time) of requests does not exceed 70ms. There are two ways to do this: either use multiple single-server queues (i.e., multiple M/M/1) or multi-server-queue (i.e., M/M/s). Use the `mm1q` simulator to investigate which of the above configurations can meet the delay requirement with lower number of servers (assume we are only allowed to use the same type of the server that the system currently uses)? Discuss the results and the way that you guess/adjust `mm1q` parameters values.

7 Question 3 (25 points)

Consider a computer system which has four processors communicating with a shared memory and a shared bus. The mean-time-to-failure of each processor is 10000 hours and follows exponential distribution. The mean-time-to-failure of the memory and bus is 8000 hour and both follow exponential distribution. What is the probability that the whole system does not fail during continuous work of 10000 hours? Calculate the probability for the following two cases:

- case 1: the system does not fail if all of the subsystems work properly.
- case 2: the system does not fail if at least one processor, the bus and the memory work properly.

8 Submission Instruction

You have to hand in your C programs, Makefile and scripts, as well as your a document to report the analytical results of the question one, and answers for question 2-3. Use `tar` to pack your files, name it [ASUID]H2 (e.g. 222222222H2), and submit it using blackboard.