IK2219/IK3506
Homework #4: Web Server Simulation

Due on 23:55, October 5th (Sunday), 2014

Jeong-woo Cho

Rules

Each student (no grouping) should carry out this homework. Although you are encouraged to discuss with each other, you have to write programming codes and solutions **in your own words**. You can hand in solutions either in a **single** Acrobat PDF file or Microsoft Word file. The length of the submission should not be more than 5 pages (font size: 10-12, paper size: A4/Letter).

If you **miss** the deadline specified on the front page of this document and hand in your solutions within exactly one week after the deadline, **30% of the total points will be taken off**, regardless of your actual score. Solutions handed in more than one week after the deadline will not be graded. Most importantly, if your solutions are incorrect, you are highly likely to receive 0 points for the corresponding questions. In this light, you should double-check if **your solutions are unequivocally correct**.

Note again that you are allowed to use only built-in functions in MATLAB and its toolboxes, *e.g.*, Statistics Toolbox. It is not allowed to use SIMULINK.

Instructions

This is the last homework of this course. In this less demanding homework, as compared with other homework, we will study the notion of *stationarity* and apply the famous *Little's Law*. In order to minimize your efforts to be taken to do this homework, it is worth while to note that the simulator you made for the previous homework can be reused as the basic block or module for this homework. That is, you just need to adapt it slightly for this homework. You would better start this homework as soon as possible because *the complexity of simulation is considerably higher* than previous ones.

Problem 1: *Extended* Little's Law

We consider a web server modeled as a single server queue, where jobs (of any type) are served a First In, First Out (FIFO) manner. The service requests from a huge number of customers are modeled as a Poisson process of intensity λ. When a service request arrives, it will generates a "type 1" job, which queues up at the end of the queue of the web server. As soon as a type 1 job is processed, it becomes a "type 2" job, which is enqueued *one more time* at the end of the **same** queue of the web server. When a type 2 job is processed, it will leave the system and the request is cleared. Note that two types of job *share* the same single FIFO queue. It is important to understand that this system is not M/GI/1 queue but a **network of two queues**, where each arrival has to be precessed twice by the server (firstly as a type 1 job, secondly as a type 2 job) and the same single queue is shared by the two types of job in a FIFO manner.

The distributions of the two types of service time are in **milliseconds** and described as follows:

- **type 1**: log-normal distribution with parameters $\mu = 2$ and $\sigma = 1$. The log-normal distribution is the distribution of the exponential of a normal random variable. The parameter set can be translated into a log-normal distribution with mean 12.18249396 millisec and standard deviation 15.96920894 millisec.
- **type 2**: uniform distribution on the interval [5 millisec, 10 millisec].

We will verify whether an **extended version of Little's Law**, which is stated in Chapter 8.2.2 (Theorems 8.2.2 and 8.2.3) holds in the network of two queues described above. In particular, we want to check the validity of Theorem 8.2.2 which states that the Little's Law can be applied not only to simplistic single queue servers introduced in Chapter 8.3 but also to **any stationary system** (or any stable system). In brief, the Little's Law must hold in the network used in this homework.

In order to verify the Little's Law, we have to measure a few averages used in its formula:

$$\lambda \text{ (mean arrival rate)} \times \bar{R} \text{ (mean response time)} = \bar{N} \text{ (mean no. of customers or jobs in the system)}$$

Problem 1: *Extended* Little's Law continued on next page...

where the word 'mean' is sometimes replaced with 'expected' or 'average' in the literature. We will measure $\bar{R}$ and $\bar{N}$ in the following. Though we can measure λ as well, we will just use the given values of λ which is in this case $\lambda = 40$ arrivals/sec (or equivalently mean inter-arrival time of 25 millisec). The simulation terminates at (simulated) time $T_s = 10000$ sec.

(a)

To begin with, we compute the mean response time $\bar{R}$ through simulations. Specifically, we measure the mean response time for type 1 job and type 2 job, which will be denoted by $\bar{R}^1$ and $\bar{R}^2$. Once again, note that 'mean response time', by definition, includes the service time. For instance, in order to measure $\bar{R}^2$, you should measure the duration of time **from** the arrival of type 2 job to the queue (right after its corresponding type 1 job is processed by the server) **to** its departure time from the system. Show the histogram of the measured response times for type 1 job and type 2 job with 50 bins. What are the *measured* mean response times, *i.e.*, $\hat{R}^1$ and $\hat{R}^2$?

(b)

Now we turn to the means of the numbers of type 1 job and type 2 job. To begin with, it is important to recall that 'mean number of jobs' is a **time** average, as compared with 'mean response time', which is an **event** average (See Question 8.3.4 in Page 249 of the textbook). However, instead of computing the time average, 'mean number of jobs', which must be sampled at arbitrary time, there is a way around it. Since the PASTA property (Chapter 7) applies to general stationary systems, (i) 'mean number of jobs' sampled just before the arrival of a Poisson process can substitute for (ii) 'mean number of jobs' sampled at arbitrary time. Since the arrival process of type 1 job in the given network is a Poisson process, we choose to use (i) and denote the means of the numbers of type 1 job and type 2 job sampled just before the arrival of each type 1 request at the queue by $\bar{N}^1$ and $\bar{N}^2$, respectively. Once again, note that they, by definition, include those who are being served. Show the histogram of the measured number of type 1 job and type 2 job with 20 bins. What are the *measured* means of the number of type 1 job and type 2 job, *i.e.*, $\hat{N}^1$ and $\hat{N}^2$?

(c)

Compare $\lambda\hat{R}^1$ and $\hat{N}^1$. Do the same for $\lambda\hat{R}^2$ and $\hat{N}^2$. Are they approximately the same?

(d)

Attach here a single MATLAB program code which generates all solutions (in particular, figures) to this problem. You don't need to commentate the code.

Problem 2: Stationarity

In this problem, we will check the range of the intensity of the arrival process λ which stabilizes the network of the two queues. In other words, what is the range of λ for the stationarity of the network? In this problem, each simulation terminates at (simulated) time $T_s = 1000$ sec (instead of $T_s = 10000$ sec in the previous problem). You are **not** required to hand in the MATLAB program used for this problem.

(a)

Plot the number of all jobs (type 1 job and type 2 job) in the queue versus time for $\lambda = 40$ and $\lambda = 60$. Does the size of the queue seem to be stabilized to a constat value as time increases?

(b)

We can now conclude safely that the upper bound of the intensity λ which keeps the network stationary or stable is somewhere between $\lambda = 40$ and $\lambda = 60$. Determine these values through running simulations for different number of $\lambda \in [40, 60]$. Can you determine this *threshold* analytically?

Before finishing this homework, make sure that there are six figures in total in your submission.