

Home assignment 1, Simulation

Let each task be in one section of the report. At the end of the report, append the code for the programs. Put everything in one **pdf** file. Also submit the source code for test runs in a zip archive where each task is separated onto folders. A report template is provided together with the assignment specification.

The purpose of this home assignment is to practice writing simulation programs using the event scheduling approach and the process interaction method. You will also study verification and validation of simulation programs.

Some general hints

When writing a simulation program using the event scheduling approach it is often of great help to start by answering the following questions:

- What variables are needed to describe the state of the system? Do not use a more complicated state description than needed.
- What events are needed?
- What shall be done when a certain event takes place? Use flow diagrams, pseudo code or write it down in words.

When writing a simulation program using the process interaction method the following questions should be answered before you start coding:

- What process classes are needed?
- What signals are needed?
- What variables are needed in the process classes?
- What shall a process do when it receives a signal?

Usually one has to iterate these questions several times. Remember that minutes spent on these questions can save hours of work later on! When you have answered these questions, the translation to a simulation program is (in principle) straightforward.

Task 1 (Event scheduling)

Assume that we have two queuing systems, let us call them Q1 and Q2. All customers first arrive to Q1. Q1 has 10 queuing places and one server. If a customer arrives to Q1 and all queuing places are occupied, the customer is rejected and never returns. When a customer has been served in Q1, it continues to Q2. Q2 has an infinite number of queuing places and one server. The times between the arrivals to Q1 are constant. The service time in Q1 is exponential with mean 2.1 seconds and the service time in Q2 is constant and equal to 2 seconds.

Write a simulation program and use it to answer the following questions:

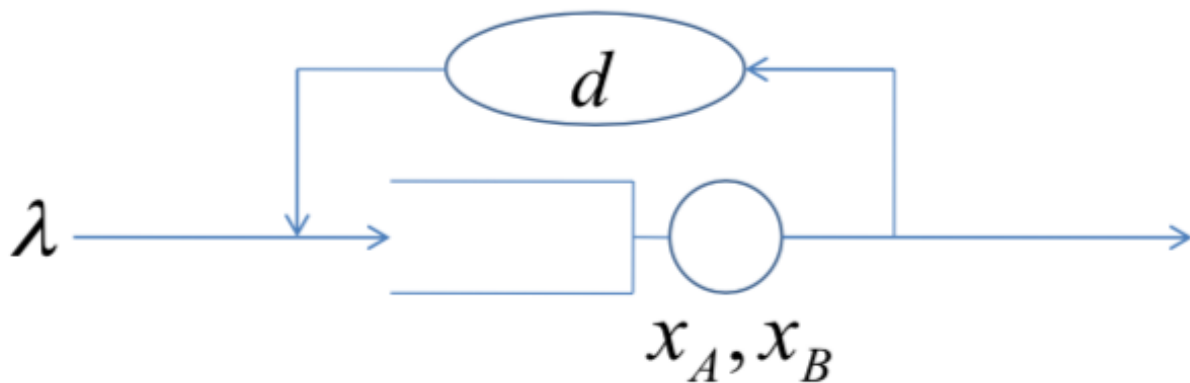
Find the mean number of customers in Q2 for the following interarrival times to Q1: 1, 2 and 5 seconds.

2. For interarrival times in question 1 you should also find the probability that a customer is rejected at Q1.

Let the times between measurements of the number of customers in Q2 be exponentially distributed with mean 5 seconds and make at least 1000 measurements.

Task 2 (event scheduling)

We shall study a model that can be used to investigate the capacity of the control processor in a switch in a network. A request for the establishment of a new virtual connection that arrives to the switch generates a job of type A. If that job has to wait for other jobs it is put in a buffer. Finally the job is executed and the connection is established. When the connection shall be taken down, a new job is generated. The jobs that tear down connections (jobs of type B) have a higher priority than jobs of type A. Thus, when a job has been served, one first looks if there are any jobs of type B in the buffer. If there are the processor starts to serve one of the jobs of type B. If there are no jobs of type B in the buffer, the processor can start to serve jobs of type A if there are any. A job that is being served is never interrupted. The model is illustrated in the figure below:



We assume that requests for the establishment of a connection arrive to the switch as a Poisson process of rate λ , i.e. the times between arrivals are exponentially distributed. Each request generates a job of type A. This job represents the work that must be done at the establishment of a connection and has a constant service time x_A . After that the job is put in a delay of constant length d that represents the lifetime of the connection. After the delay the job is put in the queue again, now as a job of type B. Finally it gets one more service, which models the teardown of the connection. This second service time is constant with length x_B . After the second service the job leaves the system.

We assume that the buffer has an infinite number of places. The parameters have the following values: $\lambda = 150 \text{ s}^{-1}$, $x_A = 0.002 \text{ s}$, $x_B = 0.004 \text{ s}$ and $d = 1 \text{ s}$.

You shall write a simulation program for this model and answer the questions below. In all questions, let the time between measurements be 0.1 seconds and collect 1000 samples.

1. Find the mean number of jobs in the buffer for the system above.
2. Let the delay distribution be exponential instead of always having the same value, but let its mean still be 1 s. What is now the mean number of jobs in the buffer?
3. Let the distribution be of constant length = 1 s again. Change the priorities so that jobs of type A have the higher priority. What is now the mean number of jobs in the buffer?
4. Explain the differences! Try to explain the results by reasoning around what happens in the system. It might help to plot the number of customers as a function of time using matlab, pyplot, Gnuplot or some other graphing tool.

Task 3 (event scheduling)

In this task you shall compare results from simulation programs to analytical results obtained by queuing theory. We shall study two connected queuing systems (a queuing network) as shown below:



Both queues have an infinite number of buffer places, i.e. customers are never rejected. The service times in the queues have an exponential distribution with mean 1. We also assume that the times between arrivals to the first queue are exponentially distributed (the means are given below). Assume the following:

x = mean time between arrivals

T = the mean time of a customer from the arrival to the first queue to the departure from the second

N = mean number of customers in both the queues

Then one can derive the following formulas:

$$N = \frac{2}{x - 1} \qquad T = \frac{2x}{x - 1}$$

Do the following:

1. Modify the simulation program written for the first task in this home assignment so that it simulates this system instead.
2. For the mean arrival times 2, 1.5 and 1.1 simulate the system and measure:
 - a. the mean number of customers in the queuing network
 - b. the mean time a customer spends in the queuing network
3. Compare the simulation results to the formulas above!

Task 4 (Process Interaction)

In this task you will simulate a Brazilian post office. In Brazil, pensioners (people over 60), pregnant women and some other groups are allowed to skip the queue in many places including post offices. This is regarded as a natural gesture of gratitude and reverence for people who need and deserve it. In this task, your mission is to answer some questions around the post office queue and help the general manager who is about to rebuild the office space and dimension the staffing.

Assume the following:

The plan is to have a queue and one teller serving customers. Customers arrive at the post office with the average inter arrival-time 5 minutes. The inter-arrival time is exponentially distributed.

The service time is also exponentially distributed and with an average of 4 minutes.

Your task is now to make some of the customers belonging to the special group that skips the queue and goes first in line and study what happens to the other people in the queue. Therefore, vary the probability of a customer arriving at the post office as follows: (10%, 20%, 50%). Assume that the probability is the same for every customer that arrives.

1. What is the average queuing time for people belonging to the two groups in each case?

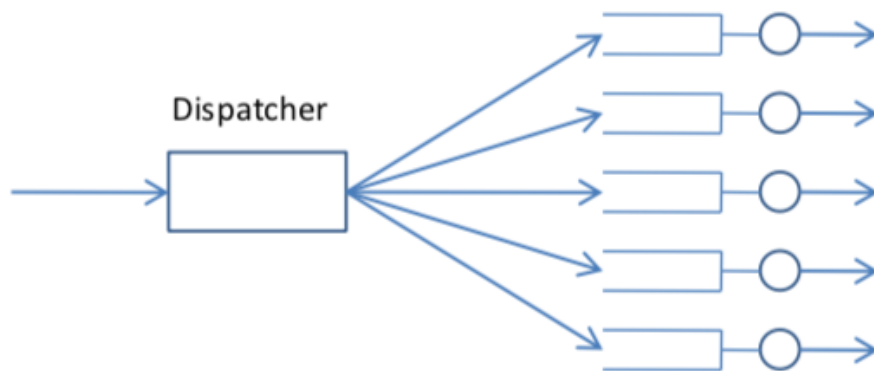
2. If the office manager decides that no customer should wait for more than 15 minutes, does he have to hire more than one teller and build two cashiers (caixas).

Simulate your system for at least 1000 arrivals when answering the questions but try 20, 50, 100, 1000 and see if you find a difference in the results.

Task 5 (process interaction)

Now you will expand on the simulator you wrote in the previous task.

You shall study load sharing among a number of service stations. Examples of this can be found in web servers, communication networks and logistic systems (including caixas). Another common application is the system to go through a number of parallel queues at security check at an airport. Each service station has a server and a buffer. All items that arrive to the system first come to a dispatcher. The dispatcher chooses to which service station an item shall be sent. See the figure:



There are 5 queuing systems. The service times exponentially distributed with mean 0.5 second in all the queuing systems. The dispatcher can choose to which queue it shall send a new job to in one of the following ways:

- i. Random, which means that with equal probability one of the queuing systems is chosen.
- ii. Round robin, which means that the first customer is sent to queuing system 1, the second to 2... the fifth to 5 and after that the next one is sent to 1 etc.
- iii. We send the job to the queuing system with the smallest number of jobs, if there is more than one system with the smallest number of jobs one of them is chosen randomly with equal probability.

The time between the arrivals to the dispatcher have a uniform distribution. The time used by the dispatcher to choose between the queuing systems is very small so we assume that it is 0.

You shall do the following:

1. Write a simulation program for the system above. All three algorithms for load distribution above shall be implemented.
2. Find the mean number of jobs in the queuing systems for all the algorithms and the following mean arrival times to the dispatcher: 0.11, 0.15 and 2.00 seconds. Which is the best algorithm?

Simulate the system for at least 100 000 seconds.

Task 6 (any method)

Assume that we have a system that consists of five components; each one of them has a uniformly distributed life length in the interval from 1 to 5. We also assume that if component 1 breaks down, also component 2 and 5 breaks down and if component 3 breaks down also component 4 breaks down. There are no other dependencies between the life lengths of the components. The system works as long as at least one component works. Find the mean time until the system breaks down. Simulate at least 1000 “runs” of the system.