

Solution to COMP9334 Revision Questions for Week 4B

Question 1

The mean response given by the M/M/1 theory is $\frac{1}{\mu - \lambda}$. For the given values of λ and μ , we expect the mean response time to be 3.3333s.

The Python file *week05A_q1.py* contains a script which, for each value of T , performs 20 simulation experiments (or replications). This is done by calling the function *sim_mm1_func()* 20 times for each value of T .

The results are plotted in Figure 1. The horizontal axis is the value of T and the vertical axis is mean response time. The horizontal line shows the value given by the M/M/1 theory. The circle shows the mean response time given by the simulation experiments. You can see that every simulation run produces a slightly different value of mean response time. However, the values appear to be scattered around the mean response time given by the M/M/1 theory. You can graphically observe that the standard deviation becomes smaller with increasing T .

The mean and standard value from these simulations are:

	T = 1000	T = 5000	T = 10000	T = 50000
mean	3.4469	3.3398	3.3742	3.3399
standard deviation	0.5197	0.3088	0.2604	0.1227

Remark: If you want to reproduce exactly the above results, you need to know the state (similar to a seed) of the generator that I have used to generate the random numbers. The setting is stored in *week05A_q1_rand_state.p*. If you use this given state (the file *week05A_q1.py* contains comments to tell you how to do this), you will be able to reproduce the results above.

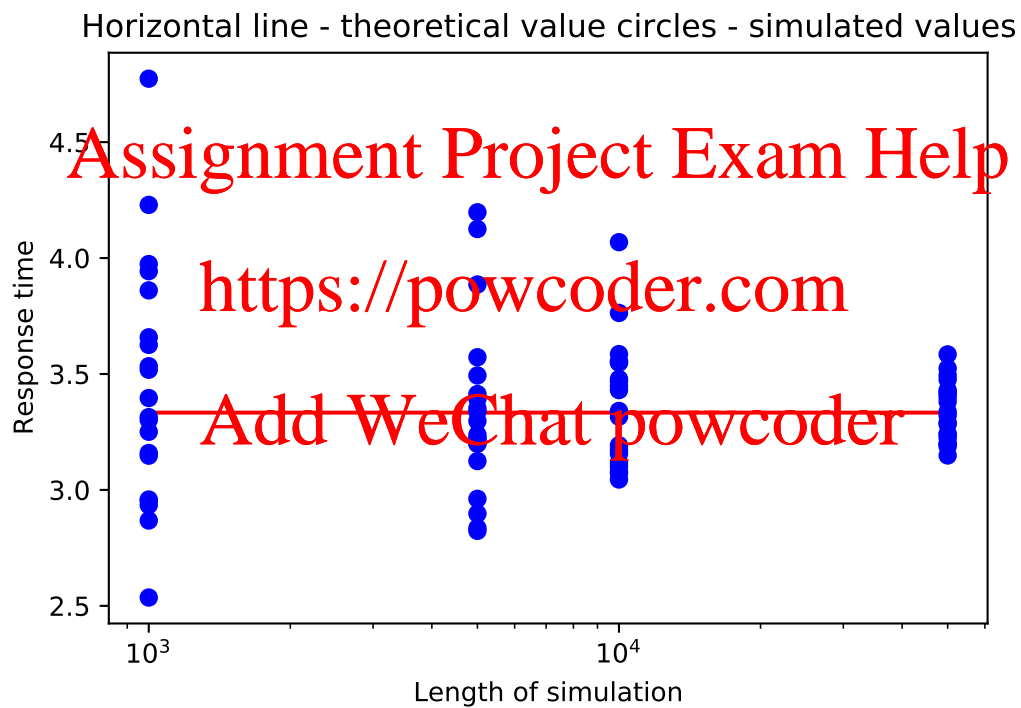


Figure 1: Response time from M/M/1 simulations of different length of time.

Question 2

Recall that in simulating a single-server queue, we maintain the following *events* and *variables*

- Events
 - Arrival to the system
 - Departure from the server
- Variables
 - The status of the server: either idle or busy
 - The number of customers in the buffer and their attributes

For an m -server queue, you will need to keep track of the departure from each of the m servers. In addition, you will need to keep track of the busy/idle status of all the m servers. These are the two main alternations that you need to make to adapt a single-server simulation program to a multi-server simulation program.

A program called *sim_mmm_lib.py* has been written to simulate an M/M/m queue.

Simulation experiments have been carried out to simulate these two queues.

1. The mean response time of an M/M/1 queue with $\lambda = 0.9$ and $\mu = 1$.
2. The mean response time of an M/M/2 queue with $\lambda = 0.9$ and for each server, $\mu = 0.5$.

Each configuration is simulated 10 times. Each simulation is carried out using an independent set of random numbers. In other words, altogether of 20 set of independent random numbers are used in the simulation.

The results of the simulation comparing the two configurations have been plotted in Figure 2. The figure also shows the mean response time predicted by the M/M/1 and M/M/2 queueing theory.

Five experiments suggest that Configuration 1 is better and the five experiments suggest that Configuration 2 is better. It does not seem to be possible to conclude which configuration is better from the simulation results.

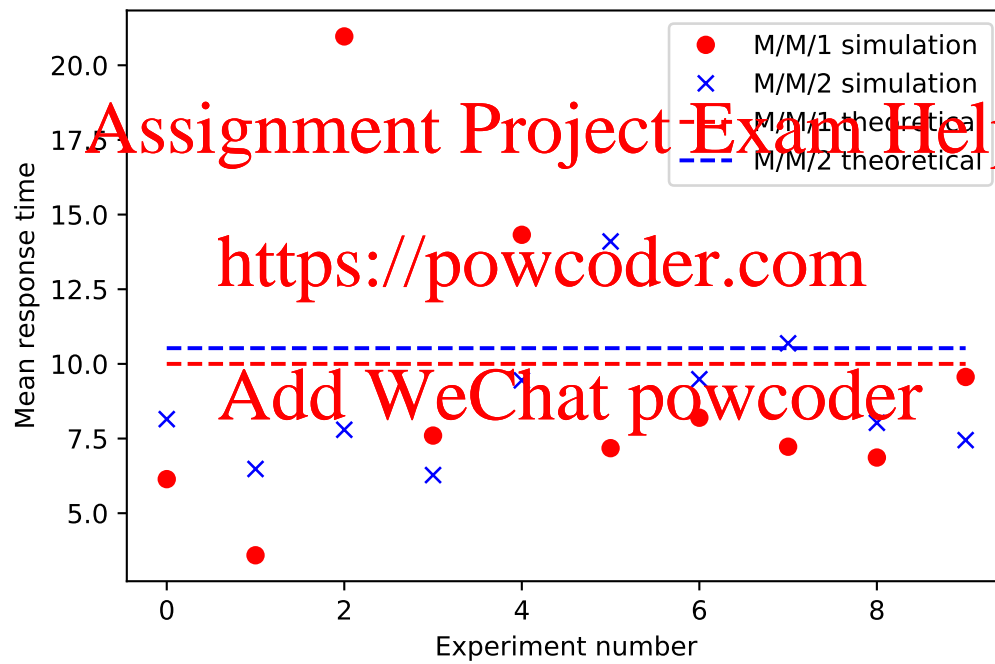


Figure 2: Comparison of an M/M/1 queue with $\lambda = 0.9$ and $\mu = 1$ and M/M/2 queue with $\lambda = 0.9$ and $\mu = 0.5$.