Technical University of Crete
Electrical and Computer Engineering

Large and Social Networks: Modeling and Analysis
TEL 422
Programming Project Report

Team 10:

Ιωαννίδης Χρήστος (2018030006)

Παπαματθαιάκη Ηλέκτρα-Δέσποινα (2018030106)

The goal of this programming project was to simulate in Matlab or Python some standard queueing systems and compare their performance, both with respect to theoretical values, as well as between themselves. This project is implemented in Python.

Part 1:

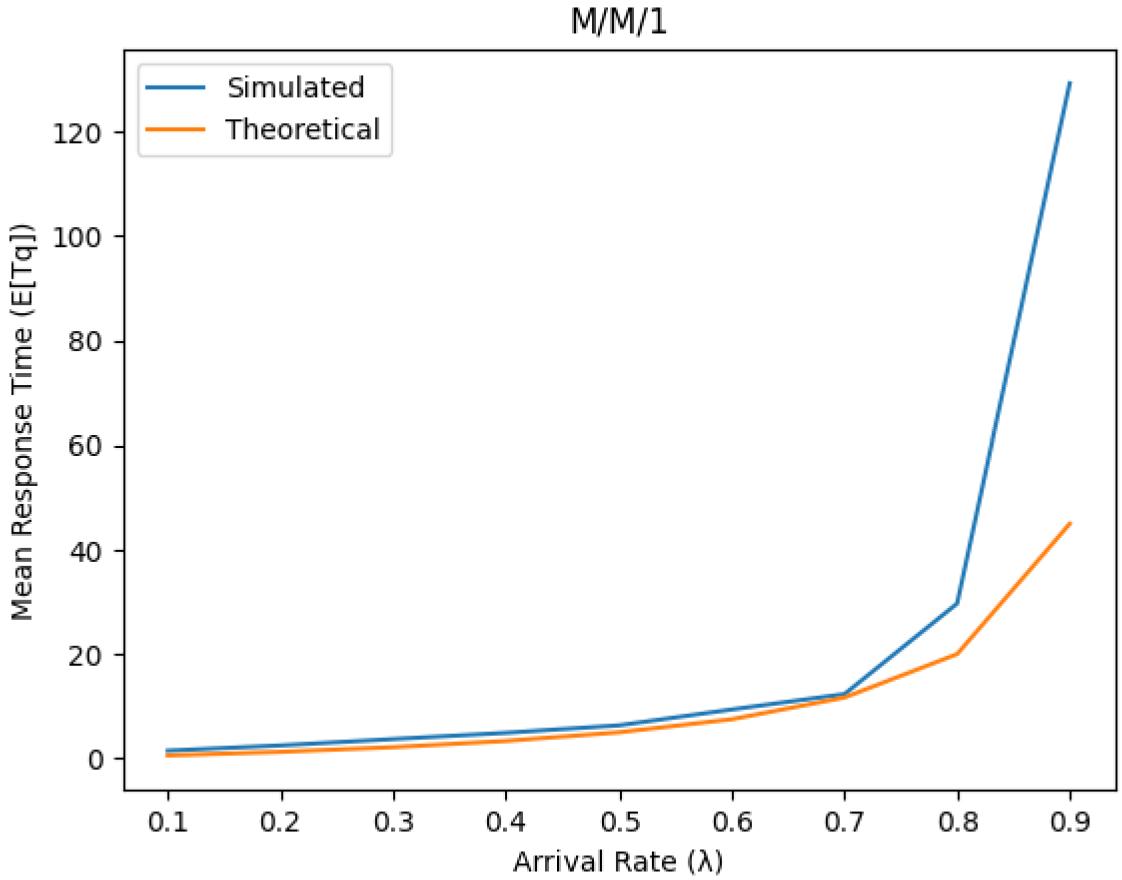
To simulate the M/M/1 Queue, a job class was created to represent a job, encapsulating its entry time into the system and processing time. Additionally, a poisson_arrival_times function was implemented to generate an array of arrival times using Poisson distribution. Moving on to the simulation part, a loop is used to iterate over ρ values ranging from 0.1 to 0.9. Within each iteration, an arrival process is simulated using another loop. By comparing the arrival time with the current time and checking the server's status, the simulation represents the process. Finally, the code generates a plot to showcase both the simulated and theoretical results.

The theoretical values of the M/M/1 Queue were found using these formulas:

$$\lambda = \rho \cdot \mu$$

$$E[T] = \frac{\rho}{\mu - \lambda}$$

The resulting graph is the following:



Observations: We can clearly see that the simulation graph has a great similarity to the theoretical, relatively small disparities occur due to the random nature of the experiment.

Part 2:

To simulate the M/G/1 First Come First Serve(FCFS) Queue a similar process was followed. The code changed on the part that the arrivals are simulated.

The theoretical values of the M/G/1 FCFS Queue were found using these formulas:

$$\mu = 0.98 \cdot 1 + 0.02 \cdot 201$$

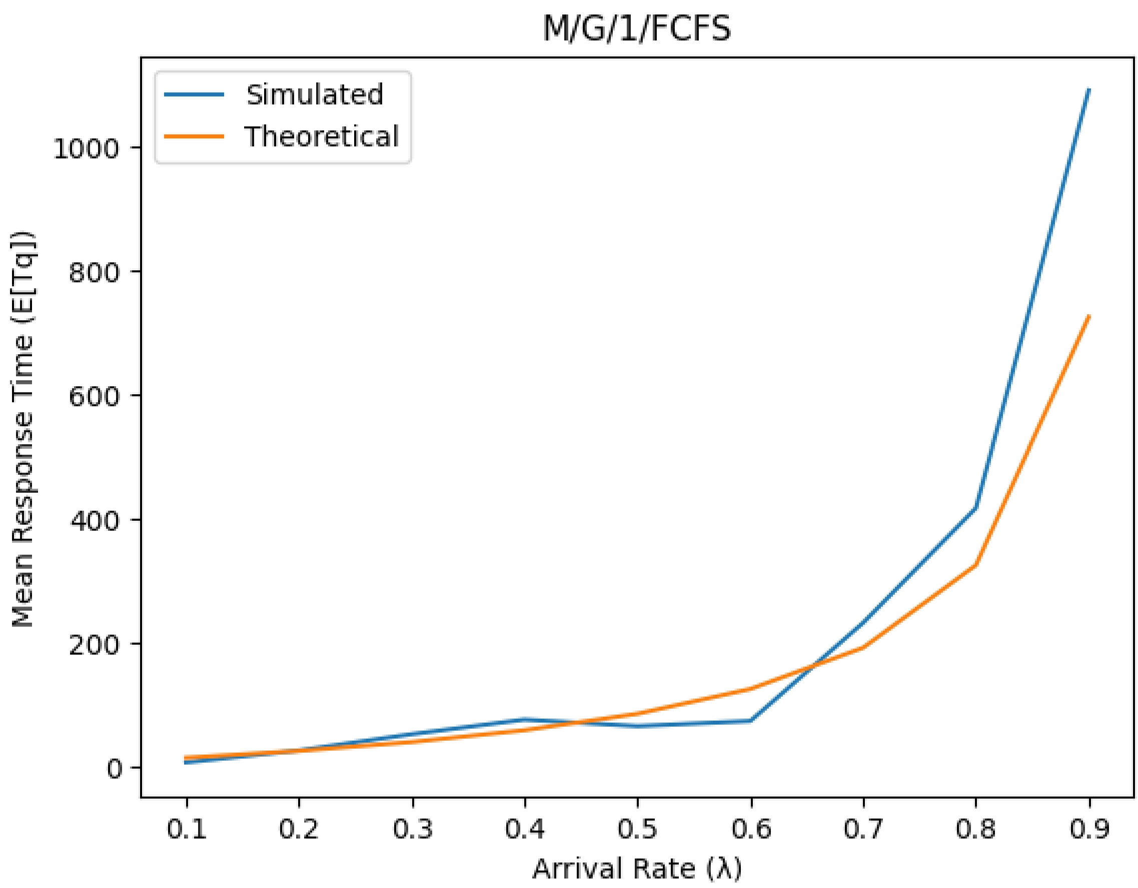
$$\rho = \frac{\lambda}{\mu}$$

$$E[S] = \mu = 0.98 \cdot 1 + 0.02 \cdot 201$$

$$Var(S) = 0.98 \cdot (1 - 0.98)^2 + 0.02 \cdot (201 - 0.98)^2$$

$$E[T] = \frac{\rho \cdot Var(S)}{2\,(1-\rho)E[S]}$$

The resulting graph is:



Observations: We can clearly see that the simulation graph has a great similarity to the theoretical, relatively small disparities occur due to the random nature of the experiment.

Part 3:

To simulate the M/G/1 Shortest Job First(SJF) Queue just like the other one, a similar process was followed. The code changed on the part that the arrivals are simulated.

The theoretical values of the M/G/1SJF Queue were found using these formulas:

$$E[S] = \mu = 0.98 \cdot 1 + 0.02 \cdot 201$$

$$Var(S) = 0.98 \cdot (1 - 0.98)^2 + 0.02 \cdot (201 - 0.98)^2$$

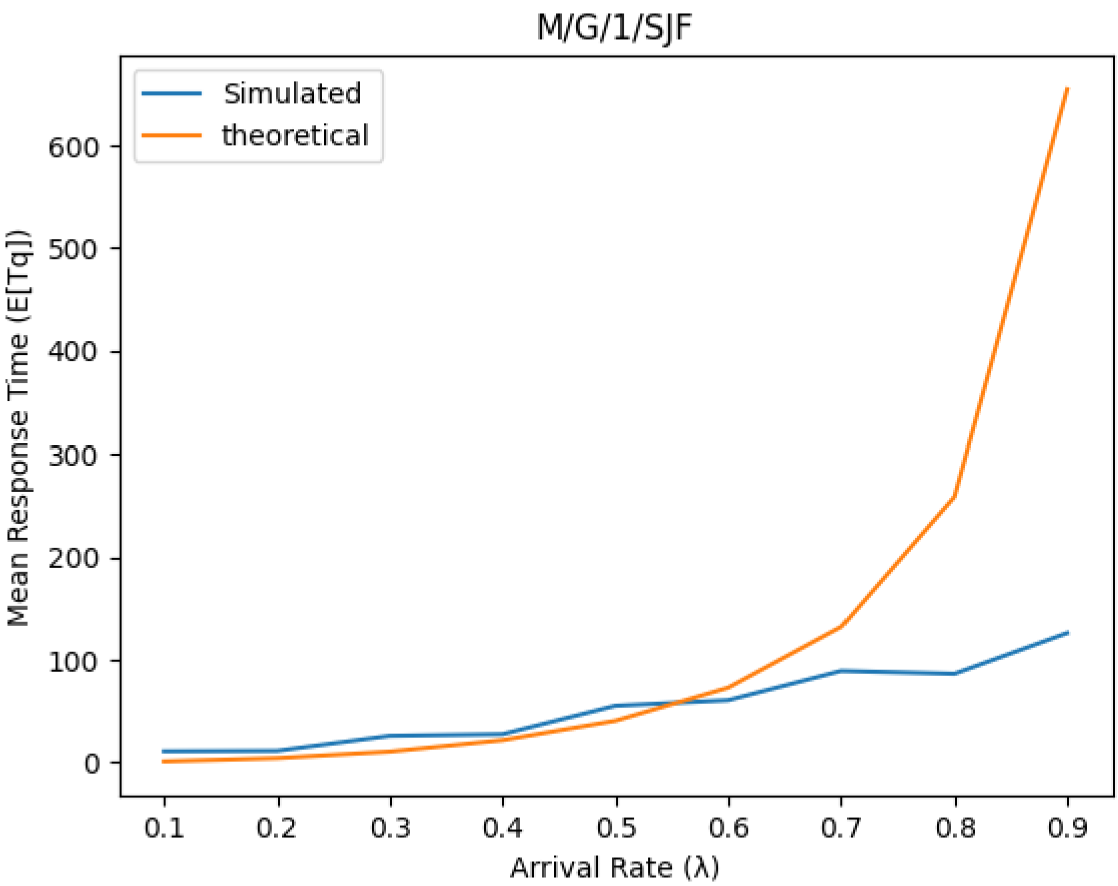
$$\rho_1 = 0.98 \cdot 1$$

$$\rho_2 = 0.02 \cdot 201$$

$$\mathrm{E[T]}_{Q1} = (\rho \cdot \mathrm{E}[S_e]\ ) / (1 - \rho_1))$$

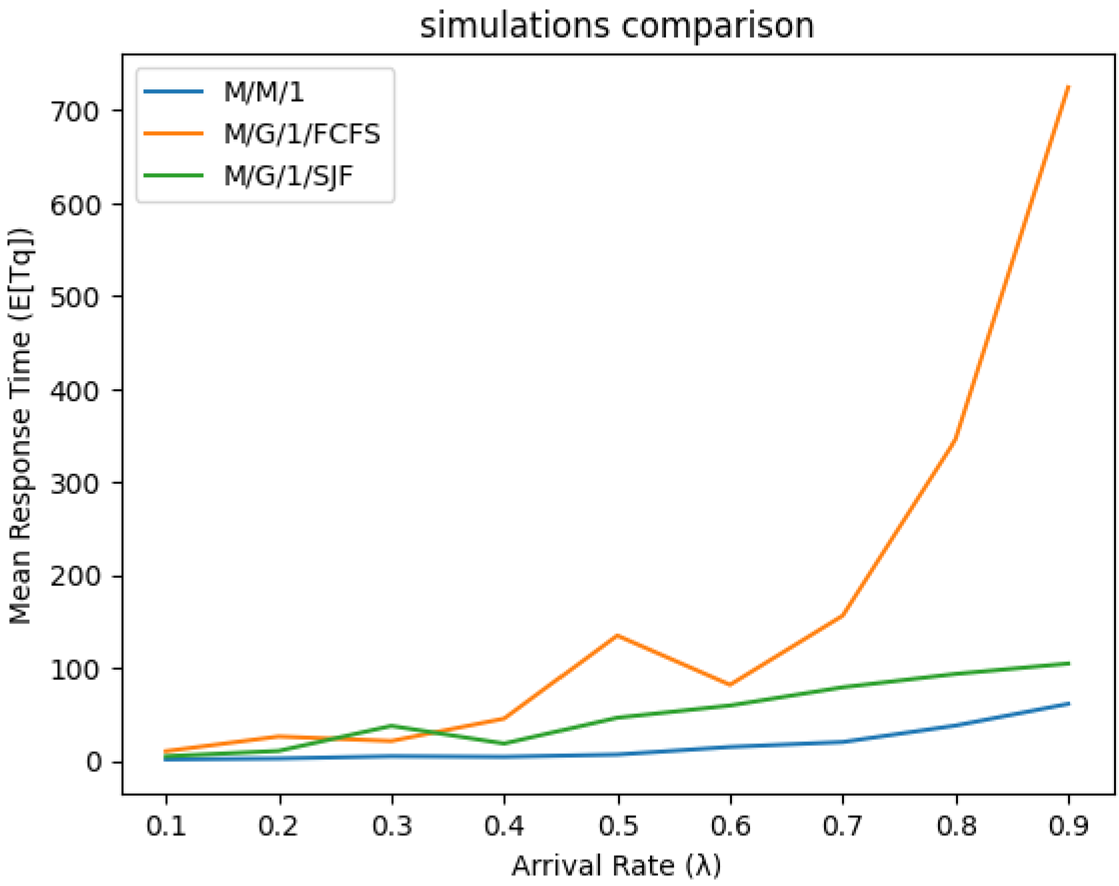
$$\mathrm{E[T]}_{Q2} = (\rho \cdot \mathrm{E}[S_e] + \rho_1 \cdot \mathrm{E[T]}_{Q1}) / (1 - \rho_1 - \rho_2)$$

$$\mathrm{E[T]} = \mathrm{E[T]}_{Q1} \cdot 0.98 + \mathrm{E[T]}_{Q2} \cdot 0.02$$



Observations: We can observe very small disparities for most values of ρ except for the higher ones, where they are probably caused because the graph has not converged properly and would probably need a lot of time to do so because of the constant flow of small jobs and also the long awaiting large jobs accumulating in the back.

Simulations Comparison:



Observations:

- The M/M/1 queue has better response time compared to the other two even though the μ is the same due to the latter ones having greater variance
- The M/G/1/SJF outperforms the M/G/1/FCFS in response time, however it is more prone to starvation for the larger jobs