An Analysis of Discrete Event Simulation models using Queueing Theory

Mehmet Bahadir Gülhan (14125250)
m.b.guelhan@gmail.com
Nicolas Schrama (11881437)
nicolas.schrama@student.uva.nl

December 6, 2021

Abstract

In this paper, we study and analyze discrete event simulations using queuing models. We set up four experiments in which we investigate the effect of (i) the distributions from which the service times are drawn, (ii) service capacity, and (iii) service queuing disciplines on the average waiting times of jobs in the queue.

First, we compare an M/M/1 queue to M/M/2 and M/M/4 queues. The simulation results suggest that average waiting times are significantly shorter when the number of servers increases, even though the inter-arrival rate of jobs increases proportionally with the number of servers. Second, we compare M/M/n queues with FIFO scheduling to M/M/n queues that prioritize jobs with the shortest service time. We conclude that priority scheduling reduces average waiting times as much as doubling the number of servers. Third, we compare M/M/n queues to M/D/n queues. The simulation results indicate that deterministic service times reduce average waiting time by orders of magnitude. Finally, we compare M/M/n queues to M/H/n queues. We conclude that waiting times are not significantly differently distributed in this case.

Key words: Discrete event simulation, queuing theory, service time distributions, service queuing disciplines

1 Introduction

For some applications, the use of continuous-time or discrete-time models does not yield accurate results. An alternative way of modelling can be used instead to improve accuracy. For example, in 2015 Akkaya and Kandemir [1] show that classical solutions of the Navier-Stokes equations lead to inaccurate predictions of flow characteristics of gases contained in micro-channels. Therefore, they use discrete event Simulation (DES) to model and predict the flow characteristics, yielding more accurate results.

In this paper, we study and analyze Discrete-Event Simulations. In particular we study queuing models. In queuing theory, a set of jobs is to be executed by servers with limited capacity. This limitation in server capacity yields a waiting line when jobs arrive at a higher rate than the system can process them.

We set up four experiments in which we investigate the effect of three factors on the average waiting times of jobs in the queue. We consider the distributions from which the service times are drawn, service capacity, and service queuing disciplines.

In section two, we provide a theoretical overview of queuing theory. Section three contains the experimental setup. Section four discusses the simulation results. Finally, we conclude in section five.

2 Theoretical Background

In this theoretical review, we discuss the main theoretical ideas driving the experiments and the results in this paper. Specifically, we elaborate on discrete event simulation, queuing theory, Kendall notation, various distributions for inter-arrival and service rates, and priority queuing.

DES is a method for modelling complex systems [2]. The main idea is to let state variables change only at discrete times. These models define the events that cause a change in the state variables and the rates at which these events occur. As a result, time in between events is not modelled, saving computational costs.

Queuing theory is a field that lends itself well to DES. It studies situations in which a population of jobs must be executed by one or more servers with limited capacity, resulting in a waiting line for the jobs [3]. Queuing theory then attempts to answer questions related to, for example, the mean waiting time in the queue, the mean utilization of the service facility, or the distribution of the number of jobs in the system.

Hence, five elements characterize a queuing system: the inter-arrival time distribution of jobs (A), the service distribution per job (B), the number of servers (m), the maximum size of the queue (N), and the service queuing disciple (S). Kendall notation summarizes these features as follows: A/B/m/N-S [3]. If N is omitted, the waiting line has no maximum size. The service queuing discipline is FIFO if S is omitted.

For the inter-arrival and service times, a few distributions are common [3]. First, the exponential distribution (M), which is the distribution of time between events in a Poisson process. A key feature of this distribution is that it is memoryless, meaning that the occurrence of the next event does not depend on any past or future information. Second, in the deterministic distribution (D), the inter-arrival or service time equals a constant value. Third, hyper-exponential distributions (H) use a weighted sum of multiple exponential distributions. This allows for tailoring of the distribution, e.g., making possible fatter tails. Fourth, in the general case (G), the distribution is not specified. We do not consider the general case in this paper.

Finally, we discuss two crucial results in queuing theory. First, a shortest job first service queuing discipline yields a lower average turnaround time than FIFO [4]. We use an example to provide the intuition. Job A arrives in queue first and requires ten seconds to complete. Job B arrives in queue second and requires five seconds to complete. Completing job A first results in a waiting time of ten second for job B, whereas completing job B first results in a waiting time of only five second for job A. Hence, completing the jobs in order of time to complete reduces waiting time more for short jobs than it increases waiting for long jobs, resulting in lower average waiting time.

Second, for FIFO scheduling, average waiting times are shorter for an M/M/n queue, server capacity μ, and inter-arrival rate $n\lambda$ than for a single M/M/1 queue with the same service rate but an n-fold lower arrival rate. We derive this result for n=2 in the appendix. We provide the intuition here. Equations 1 and 2 give the average waiting time for M/M/1 and M/M/2 queues.

$$E(W_1) = \frac{\rho}{\mu(1-\rho)} \tag{1}$$

$$E(W_2) = \frac{\rho^2}{\mu(1-\rho^2)} \tag{2}$$

$\rho = \frac{n\lambda}{\mu}$ is the system load. Equating these yields that $E(W_1)$ is longer when ρ is smaller than one, while $E(W_1)$ is shorter when ρ is larger than one. Typically, ρ is smaller than one. Intuitively, in an M/M/1 queue, jobs with large execution times, i.e., outliers, cause significant waiting times for all jobs in queue. In M/M/n queues, this problem is not as harsh, as other servers can still execute other jobs. As a result, outliers cause less significant clogging of the system.

3 Experimental setup

In this section, we discuss the details of the experiments. We run four experiments. All four use a memoryless arrival rate distribution with a mean of 0.95×15. Experiment one uses a memoryless service time distribution with a mean of $\frac{1}{15}$. We choose these means to yield a system load close to one. A system load close to one is more interesting than lower system loads, since queues only form when jobs stochastically arrive before the previous job is finished. The probability of this occurrence is larger for system loads closer to one.

Experiment two uses a memoryless service time distribution as well but gives priority to scheduling the shortest jobs first, instead of a FIFO service queuing discipline. Experiment three uses a deterministic service time of $\frac{1}{15}$ with FIFO scheduling. Experiment four uses a hyper-exponential service time distribution with fat tails. To maintain an average service time of $\frac{1}{15}$, we use the weighted sum of the two following exponential distributions: with a probability of 0.75, service time is drawn from an exponential distribution with a mean of $\frac{1}{0.5 \times 15}$; with a probability of 0.25, the service time is drawn from an exponential distribution with mean $\frac{5}{2\mu}$.

Furthermore, in our experiments, we conduct m simulations per experiment, in which k jobs are executed. In each experiment, these m simulations are conducted for one, two, and four servers. To determine the choice of m and k we look at the distribution of the waiting times for $\rho = 0.1$ and $\rho = .95$. The reason we do this is that even if the average waiting times can be analytically calculated, it can only be done in a steady state [3]. In the case of our simulations, however, we start out with an empty system. Hence, the first data points are heavily influenced by the underlying distribution of M/M/n. In order to get average waiting times close to the theoretical average, the distributions of the m simulated waiting time averages should be centered around this theoretical value. Therefore, the following section starts with an investigation of the distributions of the m=500 simulated average waiting times with k=100,500, 1000, 10000, 50000 and 100000 jobs for $\rho = 0.1$ and $\rho = 0.95$.

4 Results & Discussion

4.1 Distributions of the average waiting times

Figures 1 and 2 present the distributions of the waiting times of 500 simulations for different numbers of jobs for $\rho = 0.10$ and $\rho = 0.95$. The higher the number of jobs, the closer the distribution centers around the theoretical average. Figure 1 shows that for a low ρ, this number of jobs does not need to be high. For a high ρ value, however, Figure 2 shows that even for $k = 100000$ the distribution is still relatively wide. This means that the number of jobs is ideally even larger than 100000 in order to get very accurate average waiting times in the simulations.

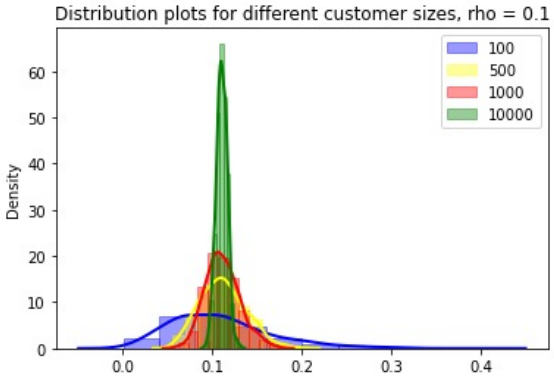

Figure 1: The distribution of the average waiting times for k=100,500,1000 and 10000.

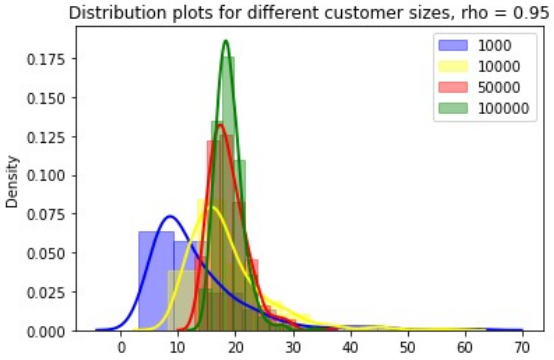

Figure 2: The distribution of the average waiting times for k=1000,10000,50000 and 100000.

Because the simulations we run for the experiments use a high ρ, the amount of jobs must to be high. A large number of jobs will, however, require a significant computation time, especially for high numbers of simulations. Because of computational limitations, we opt for seventy-five simulations and 10000 jobs. We take into consideration that the results of the experiments may be affected by this choice.

4.2 M/M/n

In this section, we discuss average waiting times when inter-arrival times and service times are exponentially distributed. We show using simulation results of M/M/1, M/M/2, and M/M/2 queues that average waiting time decreases as the number of servers increases, even though the inter-arrival time is scaled proportionally with the number of servers.

Figure 3 shows the simulation results.

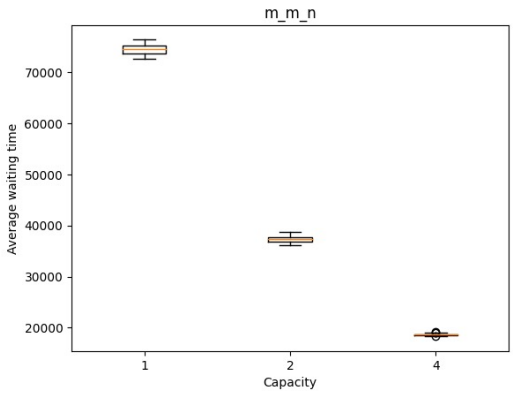

Figure 3: Average waiting times in a M/M/n queue for n = 1, 2, 4. Increasing the number of servers significantly changes the distribution of waiting times (p-values between brackets): n1 = 1 & n2= 2 (2.1e-218), n1 = 1 & n2= 4 (2.46e-44), n1 = 2 & n2=4 (1.5e-218)

. Throughout the rest of the paper, we maintain a significance level of one percent.

Figure 3 contains the box plots describing average waiting times for each number of servers (capacity). It shows that as the number of servers increases, the average waiting time significantly decreases. For example, average waiting time with one server is approximately 75000 – we specify no time unit, but this could be, e.g., 75000 seconds). Adding another server decreases average waiting time to approximately 38000 seconds, and adding another two to approximately 20000 seconds. Hence, doubling the number of servers appears to cut the average waiting time in half, at least for this range of servers.

For certainty, we also performed t-tests to confirm that average waiting times are significantly differently distributed when the number of servers varies. Figure 3's caption specifies the p-values of these tests. They confirm that average waiting times are distributed differently for different numbers of servers.

4.3 M/M/n with priority for the shortest jobs

In this section, we discuss average waiting times when inter-arrival times and service times are still exponentially distributed, but jobs are given priority based on the duration of their execution time. We compare average waiting times to those when a FIFO service queuing discipline is applied.

Figure 4 presents the simulation results.

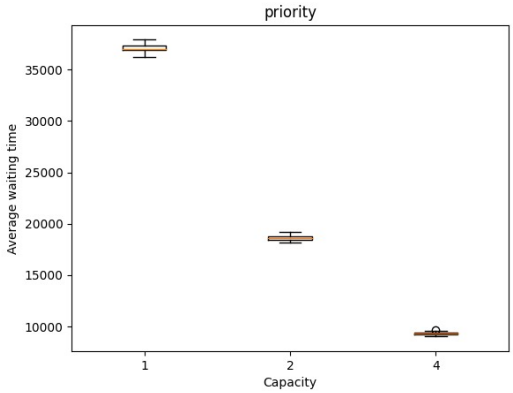

Figure 4: Average waiting times in a M/M/n queue with priority for the shortest job for n = 1, 2, 4. Increasing the number of servers significantly changes the distribution of waiting times (p-values between brackets): n1 = 1 & n2= 2 (2.4e-222), n1 = 1 & n2= 4 (1.8e-257), n1 = 2 & n2=4 (6.4e-208). Per number of servers, priority also yields a significantly different distribution than FIFO (p-values between brackets): n=1 (4.00e-213), n = 2 (4.78e-216), n = 4 (2.89e-212).

Figure 4 contains the box plots describing average waiting time for each number of servers. First, it shows that the result that increasing the number of servers decreases average waiting times also holds under a priority service queuing discipline.

Second, priority significantly decreases waiting times. Compared to a FIFO service queuing discipline, average waiting time falls from approximately 75000 seconds to approximately 37000 seconds with one server, from approximately 38000 to approximately 19000 seconds with two servers, and from approximately 20000 seconds to approximately 10000 seconds with for servers. Thus, priority seems to cut the average waiting time further in half.

To test this hypothesis, we perform a t-test with the null hypothesis that the average waiting times distribution resulting from the M/M/2 FIFO is the same as the average waiting times distribution resulting from the M/M/1 priority queue. This hypothesis is not rejected (p=0.09). We perform a similar analysis for M/M/4 FIFO and M/M/2 priority. Again, the hypothesis that average waiting times follow from the same distribution is not rejected (p=0.02) at a significance level of one percent. This indicates that priority decreases average waiting times by a similar amount as doubling the number of servers.

4.4 M/D/n & M/H/n: deterministic and hyper-exponentially distributed service times

Whereas sections four and five considered the influence of the number of servers and priority on average waiting times, this section investigates the influence of the distribution of the service times on average waiting times. Specifically, we consider deterministic and hyper-exponential service times.

Figure 5 shows the simulation results for the deterministic service times.

Figure 5 contains the box plots describing average waiting time for each number of servers. It shows that under average waiting times decrease drastically. Whereas average waiting time was in the order of tens of thousands for exponentially distributed service times, it is now in the order of tenths of seconds.

To provide intuition for this, we first consider the D/D/n case. In this case, inter-arrival and service times are deterministic. When the system load is below one, the time until the next arrival

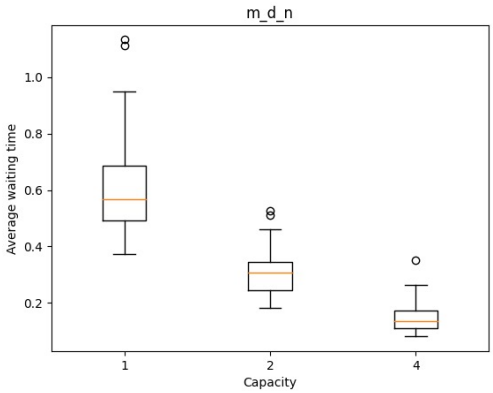

Figure 5: Average waiting times in a M/D/n queue for n = 1, 2, 4. Increasing the number of servers significantly changes the distribution of waiting times (p-values between brackets): n1 = 1 & n2= 2 (5.77e-26), n1 = 1 & n2= 4 (2.46e-44), n1 = 2 & n2=4 (4.15e-26). Per number of servers, a M/D/n also yields a significantly different distribution than a M/M/n queue (p-values between brackets): n=1 (1.89e-269), n = 2 (3.71e-273), n = 4 (3.40e-260).

is longer than the job execution time. Hence, there is no waiting time. In the M/D/n case, service time is deterministic, but inter-arrival times are exponentially distributed. Therefore, it is possible that the inter-arrival time of successive jobs is lower than the service time. This results in a queue, especially when there is only one server. Though, many jobs face no waiting time, resulting in a low average waiting time.

Finally, we consider hyper-exponentially distributed service times. Figure 6 presents the simulation results. For comparison, the M/M/n simulation results are included as well.

Figure 6 contains the box plots describing average waiting time for each number of servers for the M/H/n and M/M/n queues. It shows hyper-exponentially distributed service times yield similar results to exponentially distributed service times. This is supported by t-tests, indicating that for each number of servers the distributions of average waiting times are not significantly different (Figure 5's caption contains the p-values).

However, we expected that a fat-tailed hyper-exponential distribution would yield longer average waiting times. The intuition is that fat-tailed distributions yield more outliers. Crucially, without priority for shorter jobs, jobs with a high execution time increase average waiting more than jobs with a low execution time decrease it, especially with one server.

We do not observe this in our simulations. We consider three explanations. First, the number of jobs simulated may be too low. In section two, we show that increasing the number of jobs from 10 000 to 50 000 improves the distribution of waiting times. We opt for 10 000 jobs because of computational constraints, however. Second, the difference in average waiting time may be more expressed for system loads closer to one instead of the system load of 0.95 we use. Third, it is possible the sample size of seventy-five simulations is too small.

5 Conclusion

In this paper, we study and analyze Discrete-Event Simulations using queuing models. We set up four experiments in which we investigate the effect of three factors on the average waiting times of jobs in the queue. Specifically, we consider the distributions from which the service times are

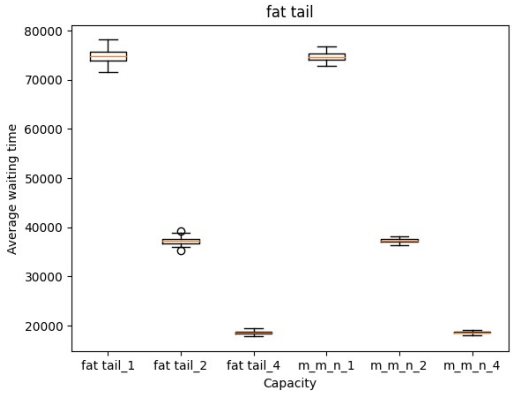

Figure 6: Average waiting times in a M/H/n queue for n = 1, 2, 4. Increasing the number of servers significantly changes the distribution of waiting times (p-values between brackets): n1 = 1 & n2= 2 (2.21e-179), n1 = 1 & n2= 4 (3.56e-211), n1 = 2 & n2=4 (1.67e-179). Per number of servers, an M/H/n queue does not yield a significantly different distribution than a M/M/n queue (p-values between brackets): n=1 (0.11), n = 2 (0.90), n = 4 (0.87).

drawn, service capacity, and service queuing disciplines.

In our first experiment, we compare an M/M/1 queue to M/M/2 and M/M/4 queues. The simulation results suggest that average waiting times are significantly shorter when the number of servers increases, even though the inter-arrival rate of jobs increases proportionally with the number of servers. We argue that systems with multiple servers suffer less from jobs with a large service time.

Second, we compare M/M/n queues with FIFO scheduling to M/M/n queues that prioritize jobs with the shortest service time. In our simulations, priority scheduling significantly reduces average waiting times as much as doubling the number of servers. We argue that prioritizing jobs with the shortest service time decreases waiting times more for jobs with a short service time than it increases waiting time for jobs with a long service time, hence decreasing the average waiting time.

Third, we compare M/M/n queues to M/D/n queues. The simulation results indicate that deterministic service times reduce average waiting time by orders of magnitude. Intuitively, when system load is below one, turning either inter-arrival time or service time from a stochastic variable to a constant value reduces the probability that a new job arrives before the old job has been completed. Hence, fewer jobs have to wait in queue, reducing average waiting time.

Finally, we compare M/M/n queues to M/H/n queues. The simulation results suggest that average waiting times are not significantly differently distributed in this case. In contrast, we hypothesized that fat-tailed hyper-exponentially distributed service time increases average waiting time. The reason is that we expected that the increase in jobs with a large service time increases average waiting time more than the increase in jobs with a short service time decreases it. We find no evidence for this hypothesis, however.

In this research we only consider some of the determinants of average waiting times. For example, we have not considered Erlang-k distributed service times. Moreover, we have not investigated the effect of alternative distributions for the inter-arrival rate. Further research in which these factors are investigated might therefore be interesting to conduct.

References

[1] Akkaya, V., R. & Kandemir I. (2015): "Event-Driven Molecular Dynamics Simulation of Hard-Sphere Gas Flows in Microchannels", *Mathematical Problems in Engineering*, vol. 2015, Article ID 842837, 12 pages, 2015. https://doi.org/10.1155/2015/842837

[2] Keeling, M. J., & Rohani, P. (2011): Modeling Infectious Diseases in Humans and Animals (First Edition), p. 201. Amsterdam University Press.

[3] Willig, A. (1999): A short introduction to queueing theory. *Technical University Berlin, Telecommunication Networks Group*, 21.

[4] Arpaci-Dusseau, R. H., & Arpaci-Dusseau, A. C. (2018): *Operating Systems: Three Easy Pieces* (1st ed.), Chapter 7, p4. CreateSpace Independent Publishing Platform.

[5] Adan, I., & Resing, J. (2002): Queueing theory.

6 Appendix

In this appendix, we derive the result that average waiting times are longer for an M/M/1 queue than for an M/M/n queue with inter-arrival times scaled proportionally with the number of servers.

We start from equation 3 for the mean number of customers in the system [5]:

$$E(L) = \frac{\rho}{1-\rho} \tag{3}$$

The PASTA property yields that the average number of customers in the system seen by an arriving customer is E(L) and that each customer has a service time with mean $1/\mu$. Hence:

$$E(S) = E(L)\frac{1}{\mu} + \frac{1}{\mu} \tag{4}$$

Together with Little's law, which states that $E(L) = \lambda E(S)$, we get the following expression for mean time spent in the system:

$$E(S) = \frac{1/\mu}{1-\rho} \tag{5}$$

The mean waiting time then follows from subtracting the mean service time from the mean time in the system:

$$E(W) = E(S) - \frac{1}{\mu} = \frac{\rho/\mu}{1-\rho} = \frac{\rho}{\mu(1-\rho)} \tag{6}$$

Similarly, for an M/M/n queue [5]:

$$E(W) = \pi_W \frac{1}{c\mu(1-\rho)} \tag{7}$$

with

$$\pi_W = \frac{(c\rho)^c}{c!}\Big(\sum_{n=0}^{c-1}\frac{(c\rho)^n}{n!} + \frac{(c\rho)^c}{c!}\Big)^{-1} \tag{8}$$

For $n = 2$, this yields:

$$E(W) = \pi_W \frac{1}{2\mu(1-\rho)} \tag{9}$$

with

$$\pi_W = \frac{2\rho^2}{1+\rho} \tag{10}$$

Hence:

$$E(W) = \frac{\rho^2}{\mu(1-\rho^2)} \tag{11}$$

Finally, since we know that $\rho < 1$:

$$\begin{aligned} \rho &< 1 \\ \rho^2 &< \rho \\ 1-\rho^2 &< 1-\rho \\ \frac{1}{1-\rho^2} &< \frac{1}{1-\rho} \\ \frac{1}{\mu(1-\rho^2)} &< \frac{1}{\mu(1-\rho)} \\ \frac{\rho^2}{\mu(1-\rho^2)} &< \frac{\rho}{\mu(1-\rho)} \end{aligned} \tag{12}$$

Thus, the average waiting time is longer for a M/M/n queue than for an M/M/2 queue. This concludes the proof.