UNIVERSITY OF AMSTERDAM

Report

Discrete Event Simulation of Queuing Systems with Variable Servers and Rates

Thursday 5th December, 2024 16:30

Student:
Lucas Keijzer - 14041073
LucasKeijzer@gmail.com
Maarten Stork - 15761770
MaartenaStork@gmail.com
Paul Jungnickel - 15716554
Paul.Jungnickel@student.uva.nl

Lecturer:
Giulia Pederzani
Andrea Tabi
Course:
Stochastic Simulation
Course code:
5284STSI6Y

Abstract

Queues are ubiquitous in computer science, business and industrial applications. When modeling a process that involves queuing, the events that are entered to a queue are often unpredictable and can be described stochastically. This report investigates the impact of different server sizes, queue disciplines and service rate distributions on the mean waiting time under varying loads. Statistical analyses, including Kruskal-Wallis and Mann-Whitney U-tests, reveal significant differences between configurations, even at the same system load. We find that larger groups of servers have lower mean waiting times. Shortest Job First queuing outperforms First-In-First-Out queuing. Modeling the job size with a hyper-exponential distribution leads to waiting times that are longer than the default exponential distribution, while a deterministic job size decreases waiting times. Our results underscore the critical role of queue management and service variability in optimizing queuing performance, particularly under high-load conditions.

1 Introduction

In service processes, when job requests exceed available servers, they are queued for later processing. To predict system performance, these processes can be modeled stochastically. This report explores queuing systems using discrete event simulation, examining the effects of server configurations and service rate distributions.

1.1 Background Theory

1.1.1 Discrete Event Modeling

Discrete Event Simulation (DES) models systems where state changes occur at specific, discrete points in time, typically triggered by events (Law, 2015). Each event represents a significant change, such as a job arriving, a server starting processing, or a job departing. Unlike continuous simulations, DES progresses by jumping between events.

DES frameworks define system states using key components: (1) an *event list* that tracks upcoming events chronologically, (2) *state variables* like queue length and server status, and (3) a *simulation clock* that advances to the next event.

This approach efficiently models queuing systems, capturing stochastic behaviors like arrivals and service times, enabling performance analysis for various configurations and distributions (Banks, Carson, Nelson, & Nicol, 2010).

1.1.2 Queuing Theory

Queuing theory provides the mathematical framework to study systems where entities, such as jobs or customers, wait in line for service (Gross & Harris, 1985; Kleinrock, 1975). It focuses on three key aspects: the (1) *arrival process*, which determines how jobs enter the system and is typically defined by an arrival rate (λ) (Medhi, 2003); the (2) *service process*, which governs how jobs are serviced, with a service rate (μ) representing the number of jobs a single server can handle per unit time; and the (3) *queue discipline*, which defines the order in which jobs are processed, such as First-In-First-Out (FIFO) or Shortest Job First (SJF) (discussed further in *Chapter 1.1.5*).

A critical performance measure in queuing systems is the *system load* (ρ), which quantifies how busy the system is. For a single-server queue ($M/M/1$), it is calculated as:

$$\rho = \frac{\lambda}{\mu}, \quad \text{where } \rho < 1 \text{ ensures stability.}$$

This metric helps evaluate whether the system can handle the incoming workload without excessive delays (Bolch, Greiner, de Meer, & Trivedi, 2006). By analyzing arrival rates, service rates, and queue disciplines, queuing theory provides insights into optimizing system performance and reducing waiting times.

Another important concept of queuing theory is Little's Law, which provides a fundamental relationship in queuing theory, connecting the average number of jobs in the system (L), the average arrival rate (λ), and the average time a job spends in the system (W) (Little, 1961):

$$L = \lambda W.$$

For the queue specifically, it is expressed as:

$$L_q = \lambda W_q,$$

where L_q is the average number of jobs in the queue and W_q is the average waiting time in the queue.

Little's Law holds under steady-state conditions, where jobs arrive and leave at consistent rates. It allows for estimating unknown metrics if two variables are known, offering a simple yet powerful validation tool.

1.1.3 Multi-Server Systems

Multi-server systems extend queuing theory to configurations where multiple servers operate in parallel, each with equal capacity (μ) (Bolch et al., 2006). The total system load in an $M/M/n$ system is distributed across n servers, with the system load now defined as:

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

These systems are more efficient at handling high arrival rates because multiple jobs can be serviced simultaneously. Multi-server configurations reduce waiting times, as parallel processing allows jobs to bypass long single queues. Furthermore, they are able to provide insights into resource allocation, highlighting diminishing returns as the number of servers increases (Hillier & Lieberman, 2008). While adding servers reduces waiting times, the improvement becomes less significant at higher server counts due to the decreased overall system load (ρ).

1.1.4 Distribution Methods

The performance of queuing systems can be significantly influenced by the choice of service rate distribution, as it affects the variability and predictability of service times. In this study, we explore three distinct service rate distributions: exponential, deterministic, and hyperexponential.

(a) **Exponential (M/M/n)**

The exponential distribution is characterized by a memoryless property, meaning the probability of a service being completed in the next moment is independent of the time already spent in service (Kleinrock, 1975). For an M/M/n queue, both the inter-arrival times and the service times are exponentially distributed. The service rate μ follows:

$$f(t) = \mu e^{-\mu t}, \quad t \geq 0,$$

where μ represents the rate parameter. This distribution is commonly used in queuing theory because of its analytical simplicity and its ability to model random, unpredictable service durations. However, its high variability can lead to unpredictable waiting times, especially in systems with high load (Gross & Harris, 1985).

(b) **Deterministic (M/D/n)**

In the M/D/n queuing model, service times are deterministic, meaning every service takes a fixed amount of time, $1/\mu$ (Bolch et al., 2006). The deterministic nature of the service time eliminates variability, which generally leads to reduced waiting times and a more predictable system behavior compared to exponential service times. The probability density function is:

$$f(t) = \delta(t - 1/\mu),$$

where δ is the Dirac delta function. Deterministic service times are advantageous in scenarios where predictability is crucial, such as automated processing tasks, as they prevent long waiting times that arise from randomness (Medhi, 2003).

(c) **Hyperexponential**

Hyperexponential distributions are used to model systems with high variability in service times, often characterized by a mixture of fast and slow jobs (Ross, 2019). In this study, the parameters were chosen so that the mean service time of the hyperexponential distribution matches that of the other service distributions, allowing for a fair comparison across all models. Specifically, 75% of jobs have an average service time of 1.8, while the remaining 25% have an average service time of 0.9. The probability density function for a hyperexponential distribution with k phases is given by:

$$f(t) = \sum_{i=1}^{k} p_i \mu_i e^{-\mu_i t}, \quad t \geq 0,$$

where p_i are the probabilities associated with each phase, and μ_i are the respective rate parameters. This distribution effectively models real-world systems with a mix of very short and very long service times, such as customer support systems where some issues are resolved quickly while others require extensive time (Trivedi, 1982). The high variability makes these systems prone to significant waiting times, particularly under heavy load conditions, making them a useful stress test for queuing models.

Each of these service rate distributions provides a unique perspective on system performance under different conditions of variability and predictability. By comparing their effects on the queuing system, we gain insight into how the nature of service times influences overall system efficiency and waiting times.

1.1.5 Scheduling Strategies

The choice of scheduling strategy in a queuing system significantly affects the order in which jobs are processed and, consequently, the average waiting time and system efficiency. This study compares two common scheduling strategies: First-In-First-Out (FIFO) and Shortest Job First (SJF).

(a) **First-In-First-Out (FIFO)**

The FIFO scheduling strategy processes jobs in the order they arrive, without prioritization (Kleinrock, 1975). Jobs are queued sequentially, and the first job to arrive is the first to be served. FIFO is simple to implement and ensures fairness, as jobs are served in the order of arrival. However, it does not prioritize, which can result in large jobs blocking smaller ones from being processed (Gross & Harris, 1985).

(b) **Shortest Job First (SJF)**
SJF scheduling prioritizes jobs with the shortest service times, aiming to minimize the average waiting time (Bolch et al., 2006). In this strategy, the next job selected for service is the shortest job by service duration. This approach reduces waiting times for smaller jobs by processing them earlier, effectively reducing the total time jobs spend in the system. However, implementing SJF requires knowledge of each job's service time in advance, which may not always be feasible in real-world systems (Trivedi, 1982). Additionally, SJF can lead to "starvation" of longer jobs in high-load systems, as smaller jobs are continuously prioritized.

1.2 Related Work

The following works provide foundational frameworks and insights necessary for understanding queuing systems, discrete event simulation, and the performance evaluation of multi-server configurations. Additional sources specific to particular aspects of the study will be cited in the references but are not discussed in detail here.

1. *Queueing Systems, Volume I: Theory* by Leonard Kleinrock (1975) provides a comprehensive theoretical foundation for queuing systems, including the mathematical models of single-server and multi-server queues. Kleinrock's analysis of system load (ρ) and its impact on performance directly informs the theoretical framework of this study (Kleinrock, 1975).
2. *Fundamentals of Queueing Theory* by Gross and Harris (1985) presents insights into the application of queuing models, with a focus on performance metrics such as average waiting time and system stability, crucial for our understanding of said topics (Gross & Harris, 1985).
3. *Discrete-Event System Simulation* by Banks et al. (2010) offers an exploration of discrete event simulation methodologies, including their use in modeling complex systems like multi-server queues, guiding the design of our simulation environment(s) (Banks et al., 2010).

These works collectively provide the theoretical and methodological foundation for our analysis of multi-server queuing systems using discrete event simulation, shaping both the simulation design and the interpretation of results.

1.3 Research Question

This study is guided by the main research question:

Do different server numbers, scheduling types, and service rate distributions in a queuing system affect the average waiting time, and if so, in what ways?

To address this, we test the hypothesis that varying these parameters significantly impacts average waiting times, compared to the null hypothesis that these variations have no significant effect.
The three parameters considered (as mentioned in the main research question) are:

1. **Number of Servers (n) in an $M/M/n$ system:**

   $$H_1 : W_q(n, \rho) \text{ decreases as } n \text{ increases.}$$

   $$H_0 : W_q(n, \rho) \text{ is unaffected by } n.$$

   Increasing the number of servers should reduce waiting times as more resources are available to process jobs. The null hypothesis assumes no effect.

2. **Queue Discipline (FIFO vs. SJF) in an $M/M/1$ system:**

   $$H_1 : W_q(\text{FIFO}, \rho) > W_q(\text{SJF}, \rho).$$

   $$H_0 : W_q(\text{FIFO}, \rho) = W_q(\text{SJF}, \rho).$$

   SJF prioritizes shorter jobs, which is expected to reduce waiting times compared to FIFO. The null hypothesis assumes both disciplines perform equally.

3. **Service Time Distributions (Deterministic, Exponential, Hyperexponential):**

$$H_1 : W_q(D) < W_q(E) < W_q(H).$$

$$H_0 : W_q(D) = W_q(E) = W_q(H).$$

   Deterministic distributions minimize waiting time variability, while hyperexponential distributions, with higher variance, should increase waiting times. The null hypothesis assumes no difference between distributions.

By systematically varying these parameters, we aim to answer these research questions. Analysis of the parameters effects on waiting times provide insights for optimizing the performance queuing systems.

2 Methods

In our numerical experiments, we ran our DES program while varying different parameters like system load, number of servers, type of queue and service time distribution. To keep the other variables constant, we choose a mean arrival time of $\mu = N$ and service time of $\lambda = \frac{1}{\rho}$. This way, we simulate roughly N events each time unit of our simulation. To obtain accurate measurements, simulations must span sufficient time for the mean waiting time to approximate its expected value. For example, an expected queue length of 100 requires simulating well over 100 events. Initial simulations are biased downward due to starting with an empty queue. To mitigate this, we measure the mean only over the final 10% of the simulation. While longer simulations reduce variance, they increase computational cost.

2.1 Comparison of Waiting Times: $M/M/1$ vs. $M/M/n$ Queues

2.1.1 Theoretical Analysis

We began by deriving the theoretical load characteristics and average waiting times for M/M/1 and M/M/N queues under FIFO scheduling. The analysis focused on understanding how system performance scales with $N \geq 2$, comparing single-server and multi-server configurations. This derivation highlights the benefits of parallel servers in managing system load and reducing waiting times.

First, we compared the M/M/1 and M/M/N queue by comparing the average waiting time (W_q) which was derived from Little's Law ($L_q = \lambda W_q$). As stated in the Introduction, the utilization factor can be calculated with $\rho = \frac{\lambda}{n\mu}$.

1. **M/M/1 Queue:** The average waiting time in the queue for an $M/M/1$ system is given by:

$$W_q = \frac{\rho}{\mu(1-\rho)}. \tag{1}$$

2. **M/M/2 Queue:** The probability that both servers are busy (P_2) is given by the Erlang B formula:

$$P_2 = \frac{\frac{\rho^2}{2!}}{\sum_{k=0}^{2} \frac{\rho^k}{k!}} = \frac{\frac{\rho^2}{2}}{1+\rho+\frac{\rho^2}{2}}. \tag{2}$$

   The average waiting time in the queue for an $M/M/2$ system is:

$$W_q = \frac{P_2}{2\mu(1-\rho)}. \tag{3}$$

Thus in the $M/M/2$ system, the waiting time W_q is reduced because the workload is distributed across two servers. For the same total load ρ, splitting the arrivals between two servers (λ/n) results in a smaller probability of both servers being busy, which leads to a shorter waiting time compared to an $M/M/1$ system. In more simplified terms, this means:

1. **Shared Load:** In an $M/M/n$ system, each server takes a portion of the workload, reducing congestion compared to the $M/M/1$ system, where a single server handles all tasks.

2. **Reduced Blocking:** In the $M/M/2$ queue, even if one server is busy, the other can handle the incoming tasks, which reduces the waiting time.

3. **Queue Dynamics:** With $n = 2$, the probability that both servers are busy is much lower than the probability of the single server being busy in the $M/M/1$ system. This reduction in the system being fully occupied directly results in lower waiting times.

2.1.2 Experimental Validation

To validate the theoretical findings, we conducted a series of discrete event simulations (DES) for n=1, n=2, and n=4 servers. Each simulation models the queuing system under varying loads to assess the relationship between server count and average waiting times. Given the stochastic nature of the simulations, multiple iterations were run to ensure statistical significance. Details of the simulation model are provided in subsection 2.3.

2.2 Comparison of Queue Disciplines: FIFO vs. SJF

To evaluate the impact of scheduling disciplines on average queue times, we compared FIFO (First-In-First-Out) and SJF (Shortest Job First) strategies (these have been explained in Chapter 1.5). By analyzing the results of these simulations, we aim to quantify the performance benefits of SJF under varying load conditions. This is done by comparing the mean waiting time for an task for different server loads. Further information on the model setup is presented in subsection 2.3.

2.3 Model

The simulation model was built using a custom discrete-event simulation framework implemented in Python with the `SimPy` library. The system models a queuing scenario with N servers, accommodating both FIFO and SJF scheduling disciplines. Key features of the model include:

1. **Arrival Process:** Job arrival times are sampled from a user-specified distribution, typically exponential, to mimic Poisson processes.
2. **Service Times:** Each job's service duration is drawn from a predefined distribution, supporting various configurations such as exponential, deterministic, and hyperexponential.
3. **Server Configuration:** The simulation supports N parallel servers, initialized as idle, with their states dynamically updated during the simulation.
4. **Queue Management:** Jobs are enqueued based on the chosen queue discipline (FIFO or SJF). The queue has a maximum capacity, beyond which jobs are rejected.
5. **Simulation Parameters:** Runs are configurable for simulation duration, random seed, server count, and verbosity to allow detailed logging.

The simulation tracks key metrics, including the number of jobs processed, queue rejection rate, average waiting time, and system utilization. Results are aggregated across multiple iterations to ensure robustness. The model's design allows flexibility in comparing server configurations and queue disciplines under varying system loads.

2.4 Comparison of Service Rate Distributions

We compared our three different service time distributions: deterministic, exponential, and hyperexponential (discussed in *Chapter 1*). Each simulation is conducted with a single server, varying system loads (ρ) between 0.4 and 0.99, and a simulation duration of 100,000 time units. To ensure reliability, 100 independent runs are performed for each configuration. The average waiting time is recorded across these runs to assess how each service time distribution influences queuing performance. This setup provides a systematic approach to analyzing differences between the distributions under identical conditions. The insights gained will highlight the impact of deterministic versus stochastic variability in service rates on waiting times.

2.5 Statistical Tests

Experimentation for our three parameters of interest involved non-parametric tests to address the potential violation of normality and the presence of skewed distributions or outliers. To evaluate whether the conditions of interest significantly differed, we employed Mann-Whitney U-tests, which are appropriate for comparing distributions without assuming normality or equal variances. For overall comparisons, we additionally conducted Kruskal-Wallis tests, a non-parametric alternative to ANOVA, which is robust for comparing medians across multiple groups when normality cannot be assumed. These tests provide reliable insights into whether variations in server count, scheduling type, or service rate distributions significantly affect average waiting times.

3 Results

3.1 Convergence behavior of the mean waiting time

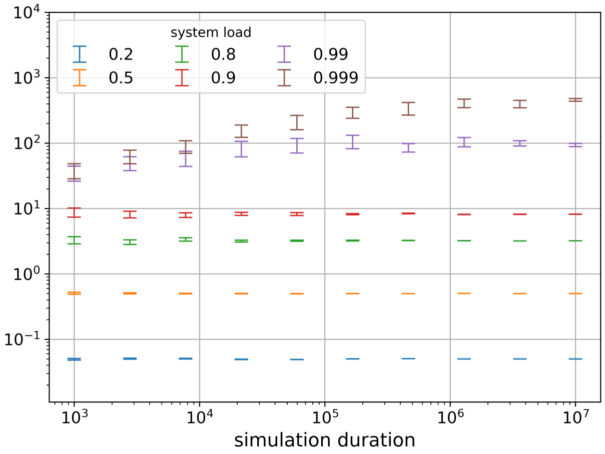


Figure 1: Convergence of the mean waiting time with simulation duration for different system loads.

Figure 1 illustrates $M/M/1$ simulation convergence under varying system loads with 95% confidence. Simulations range from 1000 runs for shorter durations to 5 runs for longer ones. At low loads, estimates converge regardless of duration. For high loads over 0.9, shorter durations significantly underestimate true values. At $\rho = 0.999$, convergence is unattainable even after 10 million events. Therefore, we limit loads to 0.99 and simulate at least 100,000 events for high loads.

3.2 Analyzing Mean Waiting Times Across Different Server Configurations

In this section, we have measured the mean waiting times with varying system loads and server counts. Figure 2 shows the mean waiting time for system loads up to 0.99.

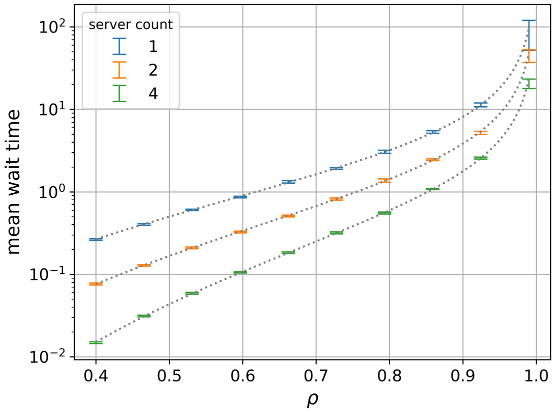
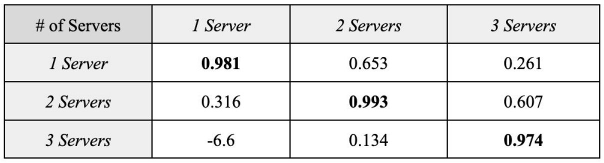


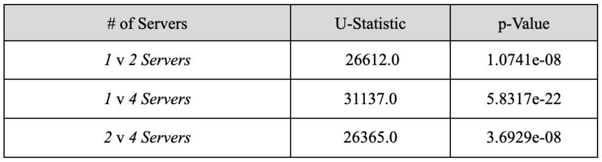
Figure 2: Mean waiting time depending on the system loads ρ for different numbers of servers. The dotted lines indicate the theoretical expected values

For these tests depicted in Figure 2, we again use the 95 percent confidence interval of our estimated mean waiting times after 20 simulation runs. To minimize the bias, we also measure the mean waiting time only in the last 10 percent of the time period. At the highest loads, we note a large increase in variance. We simulate a duration of 100.000 for the first loads and 500.000 for the highest three loads. The analytical expected value lies in the confidence interval of all our measurements. The R^2 scores of all pairs of the analytical solution and our measurements are shown in Table 1. We also perform pairwise Mann-Whitney U statistical tests, the results of which are depicted in Table 2.



# of Servers	*1 Server*	*2 Servers*	*3 Servers*
1 Server	**0.981**	0.653	0.261
2 Servers	0.316	**0.993**	0.607
3 Servers	-6.6	0.134	**0.974**

Table 1: R^2 scores of the fit quality of the analytical solution to the measured data



# of Servers	U-Statistic	p-Value
1 v 2 Servers	26612.0	1.0741e-08
1 v 4 Servers	31137.0	5.8317e-22
2 v 4 Servers	26365.0	3.6929e-08

Table 2: Pairwise Mann-Whitney U-statistics and p-values comparing the waiting times between the different server numbers.

3.3 Queue Discipline Comparison: FIFO vs. SJF

In figure 4 it is clearly visible that the SJF method which prioritizes shorter tasks consistently has a lower average waiting time for all number of system loads.

In Table 3, the values for the statistical analysis are depicted. Here, for the comparison of SJF and FIFO scheduling, a (low) p-value of 3.5758e-54 is found.

3.4 Service Rate Distributions Comparison

Figure 6 shows mean waiting times against ρ for exponential, deterministic, and hyperexponential service distributions. Hyperexponential waits grow fastest as $\rho \to 1$, while deterministic stays lowest, highlighting the impact of variability in service times.

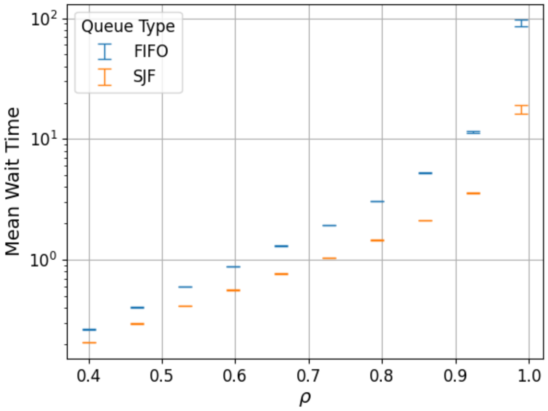
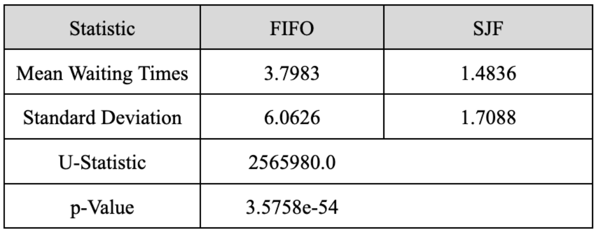


Figure 4: Comparison: FIFO vs. SJF with 100 runs of length 1000 and 1 server



Statistic	FIFO	SJF
Mean Waiting Times	3.7983	1.4836
Standard Deviation	6.0626	1.7088
U-Statistic	2565980.0	
p-Value	3.5758e-54	

Table 3: Comprehensive overview of the mean waiting times, standard deviations, pairwise Mann-Whitney U-statistics and the pariwise p-values comparing the waiting times between the different scheduling methods.

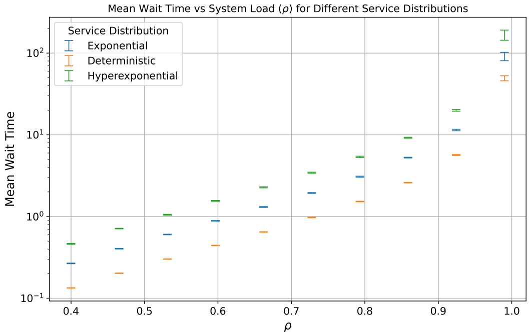
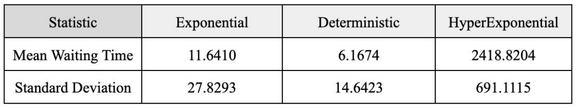


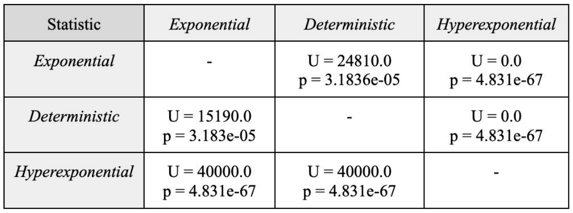
Figure 6: Pairwise Mann-Whitney U-statistics and p-values comparing the waiting times between the different scheduling methods.

In Table 4, the mean waiting times and standard deviations for Exponential, Deterministic, and Hyperexponential service rate distributions are shown. Hyperexponential distributions exhibit an exceptionally high mean waiting time and variability. In Table 5, the pairwise Mann-Whitney U-statistics and p-values for comparisons between the service rate distributions are depicted. All p-values are shown to be far below 0.05. In Table 6, the Kruskal-Wallis H-statistic and p-value for the comparison across all service rate distributions are provided. The low p-value highlights significant differences among the distributions.



Statistic	Exponential	Deterministic	HyperExponential
Mean Waiting Time	11.6410	6.1674	2418.8204
Standard Deviation	27.8293	14.6423	691.1115

Table 4: Summary of mean waiting times and standard deviations between the different service rate distributions, based on 100 simulation runs each.



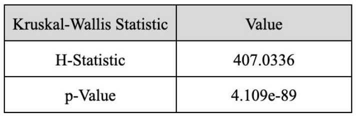
Statistic	*Exponential*	*Deterministic*	*Hyperexponential*

Exponential	-	U = 24810.0
		p = 3.1836e-05
			U = 0.0
			p = 4.831e-67

Deterministic	U = 15190.0
	p = 3.183e-05
		-	U = 0.0
			p = 4.831e-67

Hyperexponential	U = 40000.0
	p = 4.831e-67
		U = 40000.0
		p = 4.831e-67
			-

Table 5: Pairwise Mann-Whitney U-test statistics and p-values comparing the waiting times the different service rate distributions.



Kruskal-Wallis Statistic	Value
H-Statistic	407.0336
p-Value	4.109e-89

Table 6: Kruskal-Wallis F-statistic and p-value for comparing the different service rate distributions.

4 Discussion

4.1 Varying the number of Servers

From our experiments, increasing the number of servers (n) in an $M/M/n$ system significantly reduces the mean waiting time, supporting H_1. The pairwise Mann-Whitney U-tests (Table 2) reveal statistically significant differences in waiting times for all server configurations (1 vs. 2, 1 vs. 4, and 2 vs. 4 servers) with p-values well below the significance threshold ($p < 0.05$). These results confirm that increasing the number of servers provides more resources to process jobs, reducing congestion and improving system performance.

Additionally, Table 1 shows high R^2 values for the analytical fit between observed and theoretical waiting times, particularly for configurations with identical server numbers. This indicates that the analytical solution closely aligns with the experimental data, further validating the hypothesis.

4.2 Queue Discipline Comparison: FIFO vs. SJF

Our analysis confirms that SJF significantly outperforms FIFO in terms of lower mean waiting times, supporting H_1. Table 3 shows that the mean waiting time for FIFO is more than double that of SJF, with a higher standard deviation for FIFO compared to SJF. This aligns with theoretical expectations, as SJF prioritizes shorter jobs, minimizing overall waiting time. The high Mann-Whitney U-statistic and an exceptionally low p-value (both also depicted in Table 3) confirm that this difference is statistically significant at the 0.05 threshold. Thus, we are able to reject the null hypothesis (H_0) and conclude that the observed difference in waiting times between FIFO and SJF is both significant and meaningful.

4.3 Service Rate Distributions and Mean Waiting Times

Statistical analysis reveals significant differences in mean waiting times across deterministic, exponential, and hyperexponential service rate distributions, as confirmed by a Kruskal-Wallis test (Results shown in Table 6). Furthermore, pairwise Mann-Whitney U-tests (Table 5) show that hyperexponential distributions yield the longest waiting times, deterministic the shortest, and exponential intermediate (Table 4). Figure 6 further illustrates these trends visually.

These findings emphasize that service rate variability strongly affects mean waiting times, especially under high system loads (ρ), with deterministic distributions minimizing delays and hyperexponential distributions exacerbating them. This supports H_1 and highlights the importance of selecting appropriate service rate distributions in queuing system design.

4.4 Possible Improvements and Future Directions

Our analysis used static arrival and service rates, which, while practical, limits realism for systems experiencing variable demand. Extending the model to include dynamic rates reflecting real-world fluctuations, such as peak and off-peak periods, would enhance applicability. Additionally, all servers in our model were identical in capacity. Exploring heterogeneous server setups, where servers have varying service rates, could provide deeper insights into resource allocation and system optimization under more realistic conditions.

4.5 Conclusion

In response to our main research question, we can conclude that each of the examined parameters significantly affects mean waiting times. Increasing the server count reduces the waiting times, and SJF queuing outperforms FIFO queuing. Deterministic job sizes reduce waiting times compared to exponential distributions, while hyper-exponential distributions increase them.

5 Task distribution

The code for our experiments in this paper is available in the GitHub repository: `https://github.com/PaulJungnickel/Stochastic_Simulation_Assignment_2`

Author	loc	coms	fils	distribution
Lucas-Keijzer	789	11	5	22.1/42.2/26.1
PaulJungnickel	2042	26	12	56.2/40.6/52.2
MarteenStork	805	27	5	21.7/17.2/21.7

Table 2: Git-Fame

Maarten: Authored the introduction and significant portions of the theoretical background, implemented statistical tests, completed sections 1.4, general code refactoring, and added documentation. **Paul**: Contributed to the results and discussion sections, implemented the majority of the Discrete Event Simulation (DES), and handled data generation. **Lucas**: Assisted in writing the introduction, methods, and discussion, developed the Shortest Job First (SJF) queue and various metrics within the DES, and managed time handling and metrics tracking during simulations.

References

Banks, J., Carson, J. S., Nelson, B. L., & Nicol, D. M. (2010). *Discrete-event system simulation* (5th ed.). Upper Saddle River, NJ: Prentice Hall.

Bolch, G., Greiner, S., de Meer, H., & Trivedi, K. S. (2006). *Queueing networks and markov chains: Modeling and performance evaluation with computer science applications* (2nd ed.). New York, NY: Wiley-Interscience.

Gross, D., & Harris, C. M. (1985). *Fundamentals of queueing theory* (2nd ed.). New York, NY: John Wiley & Sons.

Hillier, F. S., & Lieberman, G. J. (2008). *Introduction to operations research* (9th ed.). New York, NY: McGraw-Hill Education.

Kleinrock, L. (1975). *Queueing systems, volume i: Theory*. New York, NY: John Wiley & Sons.

Law, A. M. (2015). *Simulation modeling and analysis* (5th ed.). New York, NY: McGraw-Hill.

Little, J. D. (1961). A proof for the queuing formula: L= λ w. *Operations research*, *9*(3), 383–387.

Medhi, J. (2003). *Stochastic models in queueing theory* (2nd ed.). San Diego, CA: Academic Press.

Ross, S. M. (2019). *Introduction to probability models* (12th ed.). San Diego, CA: Academic Press.

Trivedi, K. S. (1982). *Probability and statistics with reliability, queueing, and computer science applications*. Englewood Cliffs, NJ: Prentice-Hall.