Queueing theory

Stochastic Simulation: Assignment II (DES simulation)

E.L. Bakels* L.H. Bijman* S. Gijsbers*
12362980 15211312 12785985

December 4, 2023

In this paper, we investigate the properties of different queueing systems. We present a short mathematical derivation for the fact that $M/M/1$ queues have longer expected waiting times than $M/M/n$ queues with the same load characteristics. Next, we show this effect in simulations for $n = 2$ and $n = 4$. Furthermore, we show the impact of the different service queueing disciples (FIFO and SJF) and the impact of different service distributions (M, Markov, D, deterministic and H, hyperexponential) on the expected waiting time. We conclude that reordering tasks can minimise waiting time and that stochastic distributions for service time increase waiting times.

1 Introduction

Queueing theory is a branch of applied probability theory that deals with the study and modeling of queues or waiting lines in systems. It can be used to analyse the performance of different systems that incorporate queues (Cooper, 2010). Moreover, it can be used to find system designs to improve the performance of the systems, in the form of capacity planning. By analysing the queues, we can understand the behavior of the underlying model (Dudin, Klimenok, & Vishnevsky, 2020a). Evaluating the impact of an investment on the waiting times is crucial in decision-making. Therefore, models and analytical techniques are necessary to assess these scenarios (Adan & Resing, 2001).

A.K. Erlang was a Danish mathematician who developed formulas for the analysis and design of telephone systems in 1917, now known as the queueing theory. He formulated a theory to find the optimal amount of circuits and operators to reduce average waiting time. Since then this theory has been applied in multiple contexts. In 1960 it was applied to analyse the performance of time-shared computer systems, signifying its importance to computer scientists and engineers, making it possible to apply this theory to supercomputers (Cooper, 1981). Queueing theory finds applications in various

*esther.bakels@student.uva.nl, loes.bijman@student.uva.nl, sacha.gijsbers@student.uva.nl

fields such as telecommunications, computer networks, traffic engineering, healthcare, and manufacturing (Dudin, Klimenok, & Vishnevsky, 2020b). In this paper, we will analyse three different aspects of queueing theory: the number of servers in the system (n, section 3.1), in what order the customers are served (the service queueing disciple, section 3.2), and the distribution that is used to determine the service time (section 3.3). We investigate the impact of these aspects by determining the average waiting time from a simulation. For the amount of servers, we also give a mathematical derivation showing that the waiting time decreases as the number of servers increases (section 2.2).

2 Methods and theory

Within queueing theory, we discuss a system with one or multiple servers and a population of customers. The customers enter the system and are connected to a server, where they are served for an amount of time. If all servers are occupied, the customers enter a queue. The average arrival rate of the system is denoted by λ, meaning that the average inter-arrival time is given by $1/\lambda$. The capacity of n equal servers is denoted by μ and the average service time is given by $1/\mu$. The system load is defined by $\rho = \lambda/n\mu$. This number is equal to the fraction of time the servers are working. If $\rho > 1$, the queue will grow indefinitely.

2.1 Kendall notation

The Kendall notation used to classify a wide variety of queueing systems. It is given by $A/B/n/N/S$, where A denotes the distribution for the inter-arrival time of the customers, B the service distribution, m the amount of servers, N the maximum size of the queue and S the service queueing disciple. Often N, and S are omitted, in which case the queue can become infinitely long and the service queueing disciple is assumed to be FIFO (first in first out). In this study we will discuss three types of distributions for B: the exponential distribution (abbreviated M, after Markov), deterministic distribution (D), and the hyper-exponential distribution (H, see section 2.3).

2.2 Average waiting time for M/M/1 and M/M/n queues

In this section, we will prove that the average waiting times for a $M/M/1$ queue with one server, a system load ρ and capacity μ is longer than for a $M/M/n$ queue, with n servers and the same load characteristics. For both systems, we assume Poisson arrivals, meaning that the arrival times are sampled from an exponential distribution. For this proof, we will use Little's Law and the PASTA property (Poisson Arrivals See Time Averages). Little's law states that:

$$\mathbb{E}(L) = \lambda \mathbb{E}(S), \tag{1}$$

where $\mathbb{E}(L)$ is the average number of customers in the system, λ the average arrival rate of the system and $\mathbb{E}(S)$ is the average time spent in the system (Adan & Resing, 2001). The PASTA property states that for a queue with Poisson arrivals ($M/\cdot/\cdot$), the

fraction of customers finding the system in state A is equal to the fraction of time that the system is in state A. Here state A refers to a certain number of customers in the system (Adan & Resing, 2001).

We assume $\rho < 1$. Furthermore, we assume an n-fold lower arrival rate for the $M/M/n$ system than for the $M/M/1$ system. This is equal to assuming n servers. From the PASTA property we know that the average number of customers in the system seen by an arriving customer is equal to $\mathbb{E}(L)$. Furthermore, we know that the average service time is given by $\frac{1}{\mu}$. Therefore, we observe the arrival relation:

$$\mathbb{E}(S) = \mathbb{E}(L)\frac{1}{\mu} + \frac{1}{\mu} \tag{2}$$

Substituting Little's Law, we obtain:

$$\mathbb{E}(S) = \frac{\lambda}{\mu}\mathbb{E}(S) + \frac{1}{\mu} \implies \mathbb{E}(S) = \frac{1/\mu}{1-\rho},$$

We can use the average time in the system to determine the average waiting time for a queue with one server ($\mathbb{E}(W)_{M/M/1}$):

$$\mathbb{E}(W)_{M/M/1} = \mathbb{E}(S) - \frac{1}{\mu} = \frac{1/\mu}{1-\rho} - \frac{1}{\mu} = \frac{\rho/\mu}{1-\rho} = \frac{\rho}{\mu-\lambda} \tag{3}$$

Now we look at the difference between an $M/M/1$ queue and a $M/M/n$ queue. To obtain systems with the same load characteristics, we assume $\rho_{M/M/n} = \rho$ and $\lambda_{M/M/n} = \lambda/n$. We determine the average waiting time for an M/M/n queue ($\mathbb{E}(W)_{M/M/n}$):

$$\mathbb{E}(W)_{M/M/n} = \frac{\rho}{\mu - \lambda/n} \tag{4}$$

We have $\lambda/n < \lambda$ as $n > 1$, and therefore $\mu - \lambda/n > \mu - \lambda$. Thus we obtain

$$\mathbb{E}(W)_{M/M/n} = \frac{\rho}{\mu - \lambda/n} < \frac{\rho}{\mu - \lambda} = \mathbb{E}(W)_{M/M/1} \tag{5}$$

for all $n > 1$. We have hereby proved that the average waiting time for a $M/M/n$ queue is shorter than the average waiting time in a $M/M/1$ queue for all $n > 1$.

A non-mathematical explanation for this is that an $M/M/n$ queue may be more resilient to customers that require a long serving time. If a queue with a single server has a customer that takes a particularly long time due to chance, all customers behind it have to wait and the queue grows. This can create a bottleneck where all other customers have to wait a long time. In addition, tasks that take a short time may also have to wait a long time. In a queue with multiple servers, the work is distributed amongst the servers. If one customer takes a long time, another server may do tasks that take less time in parallel, which results in a less severe bottleneck, even when the system load is the same. Especially when the system load (ρ) is close to one, indicating that the servers are operating near their full capacity, a queue with $n > 1$ servers working in parallel can be more efficient to minimizing waiting times. This results in shorter waiting times compared to a $M/M/1$ queue where there is only one server.

2.3 Discrete Event Simulation

In the previous section we presented the theoretical result that the waiting times for an $M/M/n$ queue are shorter than for an $M/M/1$ queue with identical system load (ρ) and processor capacity (μ) characteristics. To verify this proof, we run a Discrete Event Simulation (DES) and simulated with a different amount of servers (1, 2 and 4 servers), with 20 evenly spaced ρ values between 0.1 and 0.99 and $\mu = 0.5$.

The DES was implemented using the `SimPy` library, a process-based discrete-event simulation framework that was implemented in Python. With this framework, an $M/M/n$ queue can be simulated using the arrival rate, capacity and number of servers. To keep the system load the same for all values of n, the arrival rate varies and is determined by $\lambda = \rho \cdot n \cdot \mu$. For the $M/M/n$ queue, we sampled the inter-arrival time and the service time from an exponential distribution. This was done with the built-in `expovariate` function from the `random` module in python. Every run was performed with 50 customers and we set $\mu = 0.5$ for all our experiments.

In the first experiment, the First In First Out (FIFO) was utilized and investigated using the $M/M/n$ queue model. This means that the customers that arrive first are the ones that are served first. This was done for $n = 1, n = 2$ and $n = 4$. The required data from the different queue's is compared with each other and a statistical test is done to see if there is a significant difference between them (see section 2.4). We expect that ρ has a lot of influence in waiting times and that the experiment provides more significant results when ρ is not much less than one. Therefore we investigated how different values of ρ influence the average waiting times. In the second experiment, we implemented SJF (Shortest Job First). SJF scheduling means that the tasks that require the shortest amount of serving time are served first, and we expect that reordering tasks this way will decrease the waiting time. In our third experiment, we implement different service rate distributions. Two other service rate distributions that will be examined are the deterministic ($M/D/n$) and hyperexponential ($M/H/n$) distribution. For the deterministic distribution, the service time is set to one value ($1/\mu$). For the hyperexponential distribution, 75% if the service times are sampled from an exponential distribution with an average of 1.0 and the remaining 25% have an exponential distribution with an average of 5.0. These numbers were chosen to obtain the same average as for the exponential and deterministic distribution with $\mu = 0.5$.

2.4 Statistical tests

In order to compare the average waiting times for different queueing systems, we will use a 95% confidence interval. We use `scipy.stats.t.interval`, which determines the confidence interval with equal areas around the median. We use the overlap of confidence intervals to obtain an indication for the minimal value of ρ for which the mean waiting times are significantly different. Furthermore, we use Welch's t-test to determine whether the difference between two mean waiting times is significant for specific values of ρ. This test does not have the assumption of equal variances, as opposed to the student's t-test. When comparing the average waiting times for different

amounts of servers, the variances tend to be different (see Figure 4). We implemented Welch's t-test with `scipy.stats.ttest_ind`, where we set the parameter `equal_var` to `False`. This test uses the corrected sample standard deviation to determine the t-statistic: $s_{\bar{X}_i} = s_i/\sqrt{N_i}$, with $s_{\bar{X}_i}$ the corrected standard deviation, and s_i, N_i the sample variance and sample size of sample i (Virtanen et al., 2020). Lastly, we use a paired student's t-test to assess the difference between low and high values of ρ for the same amount of servers. This was implemented using `scipy.stats.ttest_rel`. Here, the t-statistic is determined with the pooled standard deviation: $s_P = \sqrt{(s_{X_1}^2 + s_{X_2}^2)/2}$, with s_P the pooled standard deviation, and s_{X_1}, s_{X_2} the unbiased estimators of the population variance (Virtanen et al., 2020). By using the paired t-test we assume equal variances for the same amount of servers. One assumption of Welch's test and the paired t-test is that the variables are normally distributed. This is not the case for the (non-zero) waiting times, as is visible in Figure 1. From this Figure, it seems that the nonzero waiting times are exponentially distributed, which is in line with the theoretical result of Adan and Resing (Adan & Resing, 2001). We can violate the normality assumption if the sample size is larger than 30, which is the case for all our samples.

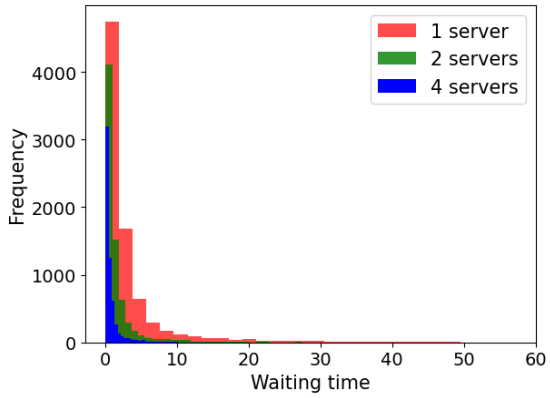


Figure 1: Distribution of the waiting times for $\rho \approx 0.94$, 100 runs, 50 customers per run, plotted on a logarithmic scale.

3 Results

Before performing our experiments, we aimed to fix the amount of runs. In order to fix this amount, we compared the width of the 95% CI of the average waiting times for different amount of runs (using $n = 2$). With every value from the list of ρ's mentioned in section 2.3, the width of the of the interval was visualized for runs between 10 and 1000 in Figure 2. We observe that the width of the confidence interval is largest for high values of ρ (close to one) and low amounts of runs. The width decreases when the number of runs increases and/or the value of ρ decreases. Over 80% of the figure has a CI with width less than 0.05, indicating that a relatively small number of runs has the same effect on the width of the 95% CI for all ρ values as the maximum number of runs tested. Balancing computing power and accuracy, we set the number of runs to 200 for all our simulations.

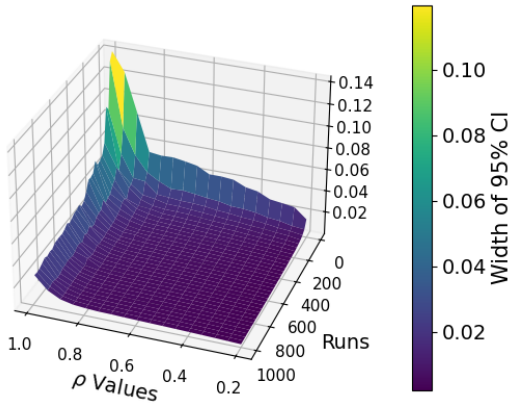


Figure 2: Width of the 95% confidence interval for different values of ρ and number of runs ($\mu = 0.5$, 2 servers ($n = 2$), 50 customers per run).

3.1 Number of servers

In our first experiment, we aim to verify the theoretical result of section 2.2 in our simulation. We compared the average waiting times and 95% confidence interval for 1, 2, and 4 servers.

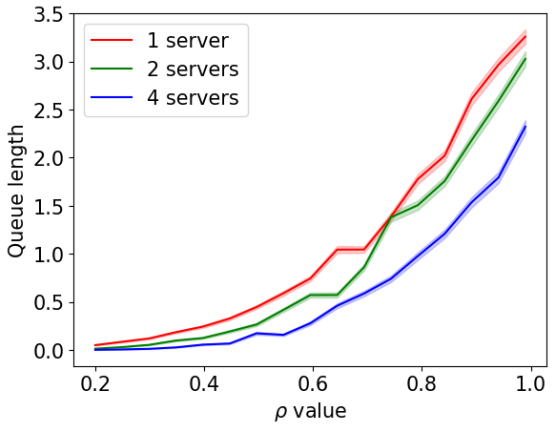


Figure 3: Average queue length and confidence interval for increasing ρ ($\mu = 0.5$, 200 runs, 50 customers per run).

The results are shown in Figure 4, when looking at the FIFO lines. As expected, a smaller number of servers lead to a longer average waiting time. Furthermore, we observe that larger ρ values result in bigger differences in waiting time than smaller ρ values. For $n = 1$, the 95% confidence interval of the line does not overlap with the confidence interval of 2 and 4 servers for any ρ value, indicating a significant difference between them. For $\rho = 0.99$, the difference in waiting time was found significant for all combinations ($n = 1$ and $n = 2$: $t(13607) = 27.424, p < 0.001$, $n = 1$ and $n = 4$: $t(10862) = 41.927, p < 0.001$, $n = 2$ and $n = 4$: $t(14393) = 27.003, p < 0.001$). When comparing $n = 2$ and $n = 4$, we observe that the 95% confidence intervals generally do not overlap, except for small values of ρ. However, a one-tailed Welch's t-test showed a significant difference for $\rho = 0.01$ ($t(10736) = 8.923$, $p < 0.001$). The observed results are in line with our expectations based on the theoretical result of section 2.2. The fact that the differences in average waiting times increase when ρ increases,

can be explained by the fact that the length of the queue increases as ρ increases, and therefore the chance of having a bottleneck increases for lower numbers of servers. We reported an increase in queue length for increasing ρ values for all tested values of n in our simulation. This result is shown in Figure 3.

3.2 Service queueing disciple (FIFO and SJF)

In the second experiment, the Shortest Job First (SJF) disciple was compared to the FIFO queue. We expected that SJF results in shorter waiting times as it prevents that short tasks have to wait for long tasks. Figure 3.2 visualizes the results and SJF performs better than FIFO queueing, especially for higher ρ values. In order to asses the presence of a significant difference, we conducted a one-tailed Welch's t-test. Before conducting the statistical test, we examined which ρ values that were relevant for investigation. To do so, we investigated for which of the discrete tested values of ρ the confidence intervals no longer overlap. For $n = 1$, we found $\rho \approx 0.319$. The difference in waiting time between FIFO and SJF for $n = 1$ and $\rho \approx 0.319$ was found significant ($t(19744) = 2.865, p \approx 0.002$). For $n = 2$, we found $\rho \approx 0.423$ ($t(19662) = 2.930, p \approx 0.002$) and for $n = 4$, we found $\rho \approx 0.577$ ($t(19044) = 2.865, p < 0.001$). The value of ρ for which the confidence intervals do not overlap is smallest for $n = 1$ and largest for $n = 4$. The SJF queueing disciple can thus lower the average waiting time in an $n = 1$ system for lower system loads than for an $n = 2$ or $n = 4$ system. Furthermore, when the system load is high, SJF decreases the waiting time by a larger factor (see Figure 4 for ρ close to one). We hypothesize that this is due to the increased length of the queue for larger values of ρ (as seen in Figure 3). When the system reaches its full capacity (ρ is close to one), the queues contain multiple customers and ordering them based on their job size can have a significant impact on the waiting time. When the system load is lower, there are less arrivals (λ is lower), the queues are shorter, and reordering tasks is not beneficial to obtain shorter queues.

As discussed in Section 2.2 and simulated in Section 3.1, the overall waiting time is shorter when the number of servers is larger. We stated that an increased number of servers is less likely to encounter bottlenecks for the entire system. We hypothesize that switching to SJF scheduling has less impact for larger values of n because there are already less bottlenecks in those systems.

3.3 Service time distribution

In the third experiment, the deterministic and hyperexponential service time distributions are compared to the Markov service time distribution from experiment 1 (section 3.1). The results are shown in Figure 5. As done in section 3.2 comparing the SJF queue with FIFO, we conducted a one-tailed Welch's t-test for the ρ values that are relevant for investigation. Firstly we compare the deterministic distribution to the Markov distribution. For $n = 1$, we report no overlap of the confidence interval for any values of ρ. For $\rho = 0.01$ we found a significant difference in waiting time ($t(13313) = 3.252, p < 0.001$). For $n = 2$, there was no overlap after $\rho = 0.16$, and at $\rho = 0.16$ there is a significant differ-

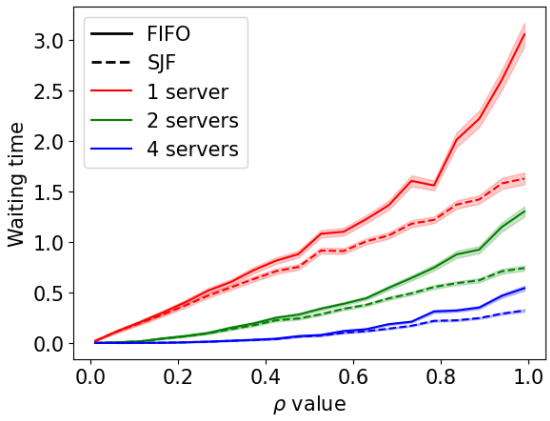
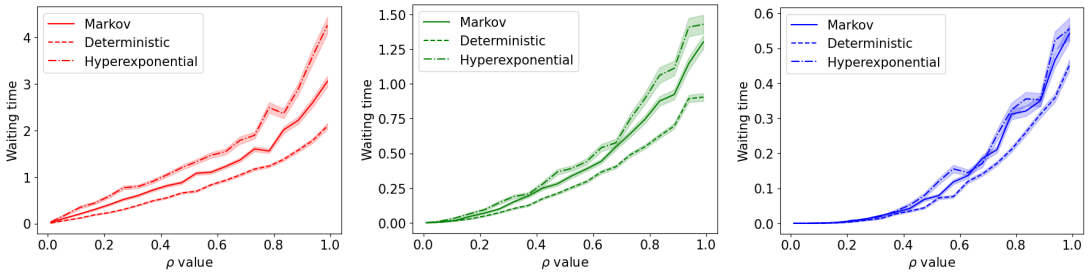


Figure 4: Average waiting times and confidence interval with FIFO (first in first out) scheduling and SJF (shortest job first) scheduling ($\mu = 0.5$, 200 runs, 50 customers per run).

ence in waiting time ($t(16264) = 5.244, p - value < 0.001$). For $n = 4$, there was no overlap after $\rho = 0.58$, with a significant difference at $\rho = 0.58$ ($t(16654) = 10.071, p < 0.001$). So for all number of servers tested, there was a significant difference between the deterministic distribution and the exponential distribution for part of the tested values of ρ. Secondly, we compare the hyperexponential distribution to the exponential distribution. For $n = 1$, there was no overlap after $\rho = 0.06$, and we report a significant difference at $\rho = 0.06$ ($t(14980) = -5.024, p - value < 0.001$). For $n = 2$, there was no overlap after $\rho = 0.42$, with a significant difference at $\rho = 0.42$ ($t(18473) = -4.255, p < 0.001$). For $n = 4$, there was overlap of the CI's of $M/M/4$ and $M/H/4$ for almost all ρ values, but did show a significant difference at $\rho = 0.99$: $t(18843) = -1.884, p = 0.030$.



(a) $M/\cdot/1$ (b) $M/\cdot/2$ (c) $M/\cdot/4$

Figure 5: Average waiting times and confidence interval for different distributions for the service time ($\mu = 0.5$, 200 runs, 50 customers per run).

The results show that the service distribution can significantly change the average waiting times of a queueing system. For larger values of ρ, the average waiting times of an $M/D/n$ queueing system are significantly lower than the waiting times of an $M/M/n$

or $M/H/n$ system (the latter by extension) for $n = 1, 2, 4$. We hypothesize that this is due to the fact that due to stochasticity, one customer can have a longer service time than average, which can increase the size of the queue during a busy period. Furthermore, we conclude that there is a significant difference between waiting times of the exponential and the hyperexponential distribution. For $n = 1$ and $n = 2$ this is the case for multiple values of ρ, whereas for $n = 4$ this is only the case for $\rho = 0.99$. We hypothesize that this is due to the fact that the average of the hyperexponential distribution and the exponential distribution is the same. We start to see this for $n = 4$, where the confidence intervals overlap for most values of *rho*. When the number of servers is small, outliers in service time have a larger impact on the overall waiting time. When there are more servers, it is more likely that the effect of one outlier (in the form of a high service time) is cancelled out by multiple customers with shorter service times than average.

4 Conclusion and discussion

We conclude that the average waiting time is shorter for an $M/M/n$ queue than for an $M/M/1$ queue. This was mathematically shown in Section 2.2 and confirmed by the simulation experiment in section 3.1 for $n = 1, 2, 4$. Furthermore, we conclude that the average waiting times with the Shortest Job First service queueing disciple are shorter than First In First Out (Section 3.2). Thirdly, we conclude that an $M/D/n$ queue has lower average waiting times as opposed to an $M/M/n$ or $M/H/n$ queue, and an $M/M/n$ queue has lower average waiting times than an $M/H/n$ queue. These difference are largest when the number of servers is equal to one. For $n = 4$, the decrease in waiting time is less. We hypothesize that this is due to the fact that stochasticity has a larger impact when the number of servers is low. One customer with a larger service time has a larger chance to increase the size of the total queue and thereby increase the overall average waiting times. With an SJF strategy this phenomenon is counteracted and therefore SJF has the largest impact on the average waiting time for lower numbers of servers.

Based on the findings of this study, several potential subjects for future research emerge. We have chosen the amount of runs in our study based on an arbitrary boundary on the width of the confidence interval in Figure 2. Testing the findings with more runs can provide more insights. Similarly, our Discrete Event Simulation (DES) had a value of 50 tasks/customers per run. This was done to prevent an infinite grow of the queue, but other values could be experimented with. In our comparisons in experiment 2 and 3 (Section 3.2 and 3.3), Welch's tests were performed for single values of ρ, and not for the interval of 0.01 until 0.99. This was done because we were interested in the ρ were the 95% CI did not overlap anymore, but this method may be improved by investigating multiple ρ values in one statistical test. Additional research extending these experiments could be done by testing waiting times for different service queueing disciples or arrival distributions. In addition to this, the ratio of improvement for SJF and FIFO seems similar for a different amount of servers, and could be further investigated. Furthermore, our experiments could be repeated for an increasing amount of servers. Another direction

could be in the development of an optimization function that minimizes the waiting time given the costs of the addition of extra servers or the computational cost to determine the job length upon entry of a customer. Finally, we have assumed n equal servers in our experiments. Future research could look into the impact of multiple servers with different capacities. This research can be tailored to the wide variety of applications of queueing theory.

References

Adan, I., & Resing, J. (2001). *Queueing theory: Ivo adan and jacques resing.* Eindhoven University of Technology. Department of Mathematics and Computing ....

Cooper, R. B. (1981). Queueing theory. In *Proceedings of the acm '81 conference* (p. 119–122). New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/800175.809851` doi: 10.1145/800175.809851

Cooper, R. B. (2010). Queueing notation. *Wiley Encyclopedia of Operations Research and Management Science*.

Dudin, A., Klimenok, V., & Vishnevsky, V. (2020a). *The theory of queuing systems with correlated flows.* doi: 10.1007/978-3-030-32072-0

Dudin, A., Klimenok, V., & Vishnevsky, V. (2020b). *The theory of queuing systems with correlated flows.* doi: 10.1007/978-3-030-32072-0

Virtanen, P., Gommers, R., Oliphant, T., Haberland, M., Reddy, T., Cournapeau, D., ... others (2020). Fundamental algorithms for scientific computing in python and scipy 1.0 contributors. scipy 1.0. *Nat. Methods*, *17*, 261–272.

5 Work distribution

Author	**Surviving Lines**	**Commits**	**Files**	**Distribution**
Esther Bakels	2234	13	8	48.2/43.3/42.1
Loes Bijman	1623	15	7	35.0/50.0/36.8
Sacha Gijsbers	779	2	4	16.8/6.7/21.1