Advanced Systems Lab (Fall'16) – Third Milestone

Name: *Taivo Pungas*
Legi number: *15-928-336*

Grading

Section	**Points**
1	
2	
3	
4	
5	
Total	

Contents

Notes on writing the report

The report does not need to be extensive but it must be concise, complete, and correct. Conciseness is important in terms of content and explanations, focusing on what has been done and explanations of the results. A long report is not necessarily a better report, especially if there are aspects that remain unexplained. Completeness implies that the report should give a comprehensive idea of what has been done by mentioning all key aspects of the modeling and analysis effort. Limited analysis because of flaws in the system or lack of experimental data from Milestones 1 or 2 are not valid arguments for an incomplete report. If bugs or lack of data prevent you from doing a correct analysis, the system must be debugged and new data collected. In case the system has been modified, include a short description of the changes as an appendix.

Remember that this is a report about modeling and analyzing the system you have designed and built, using the experimental data you have collected. There is no unique way to do the report and you may choose to focus on different aspects of the system as long as you deliver a complete analysis of its behavior. Keep in mind that, *for all queuing models in the report*, you need to explain how the the parameters of the model were determined and from which experiments the data comes from (adding a reference to the exact graph, table, etc. from the previous milestones). You have to find all system metrics that can be derived using the corresponding formulas and then match to the experimental results, explaining the similarities and differences in quantitative and qualitative terms. The calculations and the numbers you derive might need to be explained with references to the logs and sources in the previous reports. Make sure to mark these references, as well as the ones pointing to experimental results clearly. *Missing parts of the above requirements might lead to significant loss of points in each section.*

The report should be organized in sections as explained in the next pages, and each section should address at least the questions mentioned for each point. You might be called for a meeting

in person to clarify aspects of the report or the system and to make a short presentation of the work done. By submitting the report, you confirm that you have done the work on your own, the data used comes from experiments your have done, you have written the report on your own, and you have not copied neither text nor data from other sources.

The milestone is worth 200 points.

1 System as One Unit

1.1 Data

The experimental data used in this section comes from the updated trace experiment, found in `results/trace_rep3`. For details, see Milestone 2, Appendix A. The first 2 minutes and last 2 minutes were dropped as warm-up and cool-down time similarly to previous milestones.

1.2 Model

In this section I create an M/M/1 model of the system. This means the following definitions and assumptions:

- The queues are defined as having infinite buffer capacity.
- The population size is infinite.
- The service discipline is FCFS.
- Interarrival times and the service times are exponentially distributed.
- We treat the SUT as a single server and as a black box.
- Arrivals are individual, so we have a birth-death process.

Parameter estimation Using the available experimental data, it is not possible to directly calculate the mean arrival rate λ and mean service rate μ so we need to estimate them somehow. I estimated both using throughput of the system: I take λ to be the *mean* throughput over 1-second windows, and μ to be the the *maximum* throughput in any 1-second window, calculated from middleware logs. I chose a 1-second window because a too small window is highly susceptible to noise whereas a too large window size drowns out useful information.

Problems The assumptions above obviously do not hold for our actual system. Especially strong is the assumption of a single server; since we actually have multiple servers, this model is likely to predict the behaviour of the system very poorly. A second problem arises from my very indirect method of estimating parameters for the model (and an arbitrary choice of time window) which introduces inaccuracies.

1.3 Comparison of model and experiments

	variable	predicted	actual
1	mean_num_jobs_in_system	3.95	1.57
2	std_num_jobs_in_system	19.55	0.57
3	mean_num_jobs_in_queue	3.15	1.43
4	utilisation	0.20	0.93
5	mean_response_time	0.38	14.55
6	response_time_q50	0.27	12.00
7	response_time_q95	1.15	30.00

Table 1: Comparison of experimental results and predictions of the M/M/1 model.

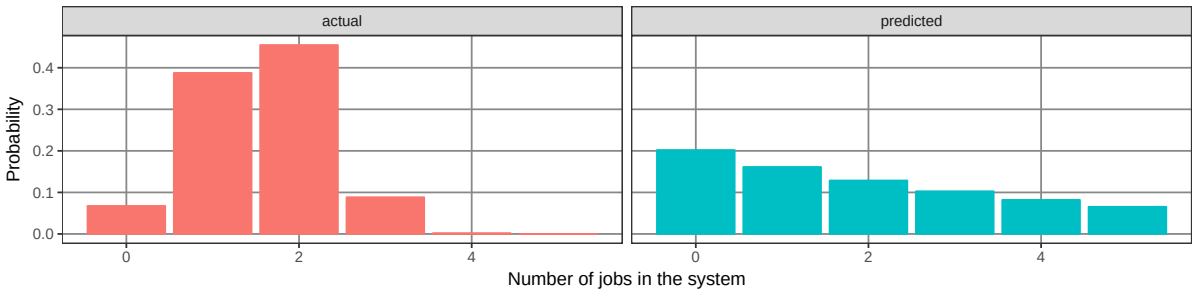


Figure 1: Distribution of the number of jobs in the system: experimental results and predictions of the M/M/1 model.

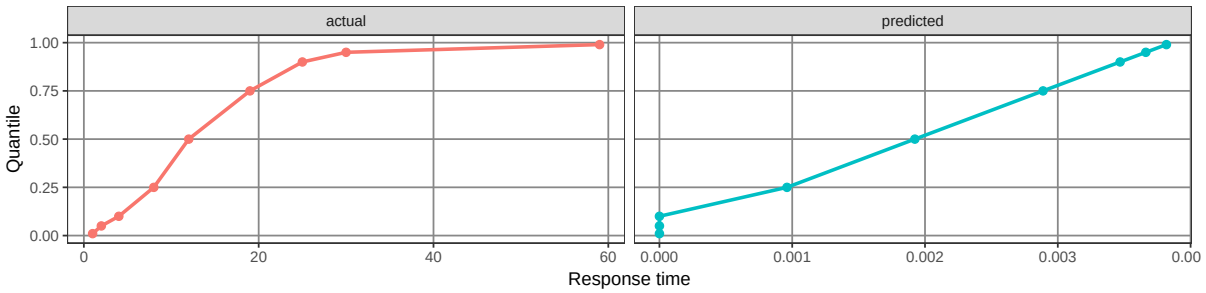


Figure 2: Quantiles of the response time distribution: experimental results and predictions of the M/M/1 model. Note the extreme difference in the response time scale.

2 Analysis of System Based on Scalability Data

2.1 Guidelines

Length: 1-4 pages

Starting from the different configurations that you used in the second milestone, build M/M/m queuing models of the system as a whole. Detail the characteristics of these series of models and compare them with experimental data. The goal is the analysis of the model and the real scalability of the system (explain the similarities, the differences, and map them to aspects of the design or the experiments). Make sure to follow the model-related guidelines described in the Notes!

2.2 Model

TODO: describe the model I was using, and how I found the parameters

bad model because I map requests to servers uniformly. M/M/m assumes that each server takes a request when it finishes with the previous one, but that is not true in my case.

2.3 Data

The experimental data used in this section comes from TODO: Section x and can be found in TODO:.

2.4 Comparison of model and experiments

3 System as Network of Queues

3.1 Guidelines

Length: 1-3 pages

Based on the outcome of the different modeling efforts from the previous sections, build a comprehensive network of queues model for the whole system. Compare it with experimental data and use the methods discussed in the lecture and the book to provide an in-depth analysis of the behavior. This includes the identification and analysis of bottlenecks in your system. Make sure to follow the model-related guidelines described in the Notes!

4 Factorial Experiment

4.1 Guidelines

Length: 1-3 pages

Design a 2^k factorial experiment and follow the best practices outlined in the book and in the lecture to analyze the results. You are free to choose the parameters for the experiment and in case you have already collected data in the second milestone that can be used as source for this experiment, you can reuse it. Otherwise, in case you need to run new experiments anyway, we recommend exploring the impact of request size on the middleware together with an other parameter.

5 Interactive Law Verification

5.1 Guidelines

Length: 1-2 pages

Check the validity of all experiments from one of the three sections in your Milestone 2 report using the interactive law (choose a section in which your system has at least 9 different configurations). Analyze the results and explain them in detail.

5.2 Data

The experimental data used in this section comes from Milestone 2, Section 2 (Effect of Replication) and can be found in `results/replication`.

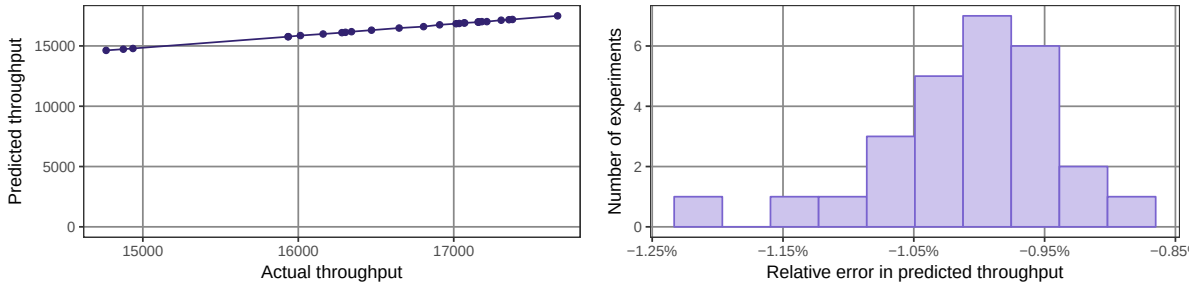
5.3 Results

which experiment I chose, did I remove beginning and end, ...

mention mean error

TODO: also show graphs as function of parameters (like in ms2 exp2)?

TODO: why do I predict lower throughput? because total cycle time is higher than it should for given throughput – but why? since measuring is done by memaslap, probably the problem lies on that side



(a) TODO:

(b) TODO: . note scale

Appendix A: Template appendix