PEA Final Project Presentation

Giovanni De Novellis - 10608672

Contents Index

Problem Description

The problem consisted in designing and analyzing the workflow of an embedded system company, with the goal of finding the optimal number of projects to work on in parallel in order to have the highest Throughput and the lowest Response Time possible.

A project always starts in the Formalize Specifications phase and then is split into two logical branches:

- In the first branch the project undergoes the Design hardware phase followed by the Breadboard Hardware phase.
- In the second branch there is the Software development phase.

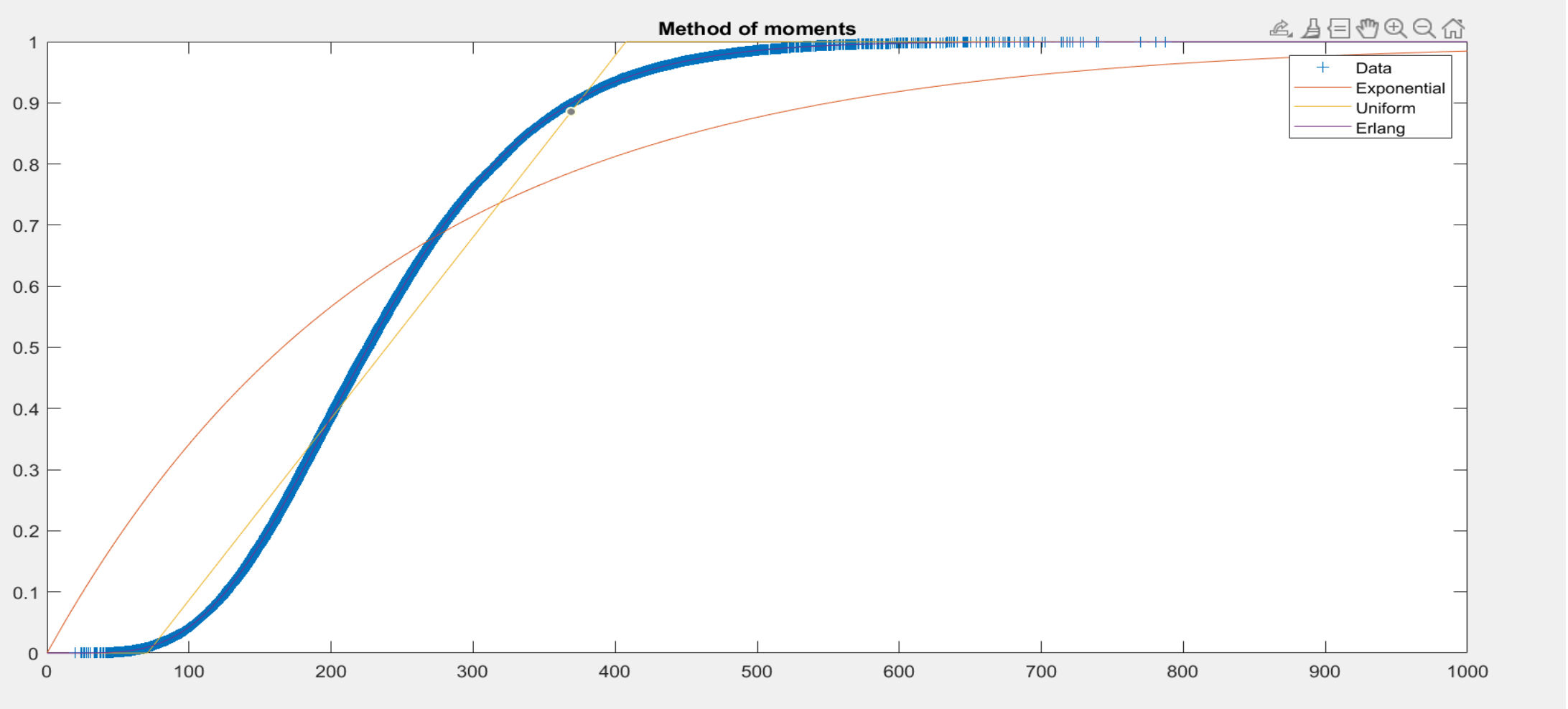
Finally, the two branches are joined into the final Testing phase.

Fitting

For each of the previously mentioned phases a file with traces for the service time in hours was given, and a fitting to characterize the most suited distribution has been done using matlab.

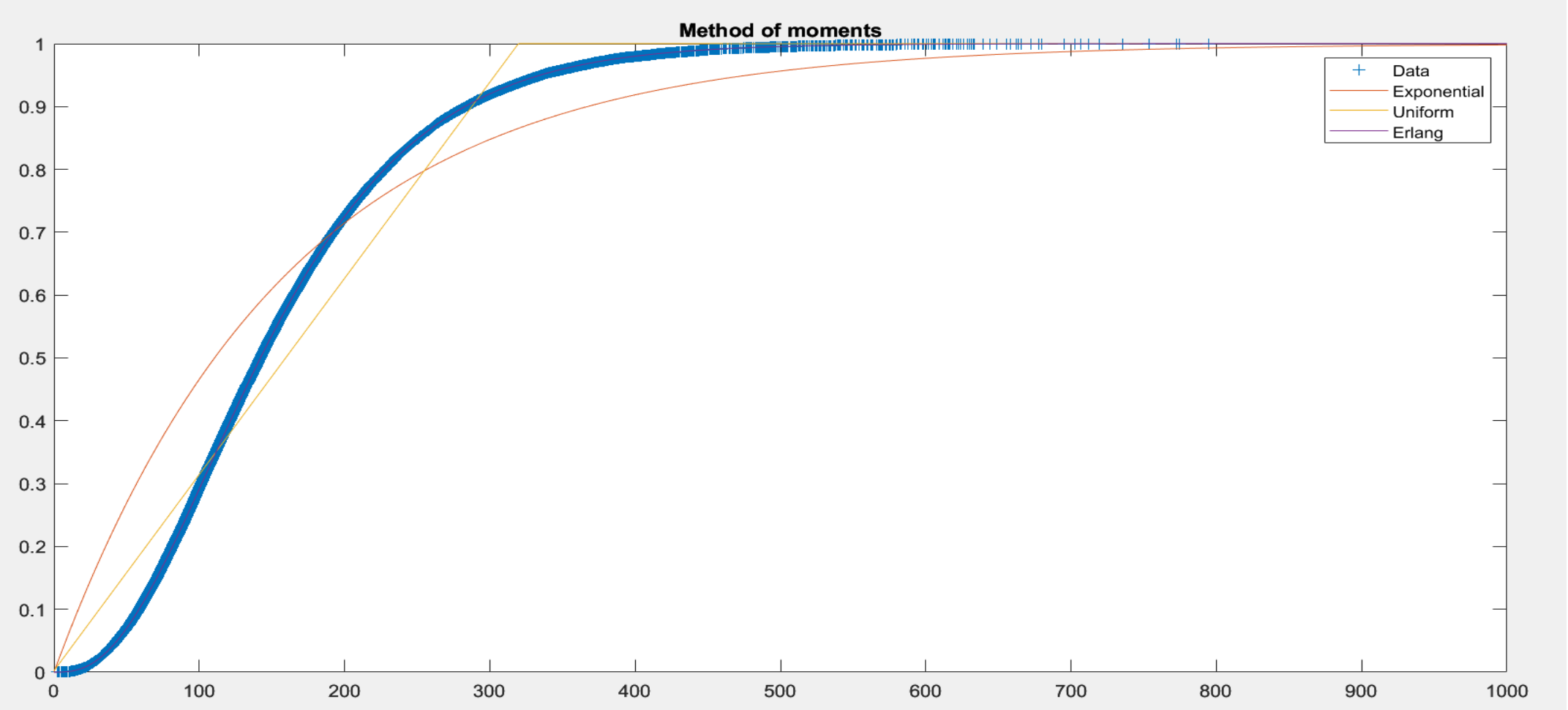
The fitting has been done using the method of moments and trying some of the distributions shown in class (Uniform, Exponential, 2-Stages HyperExponential, Erlang).

Fitting – Formalize Specifications



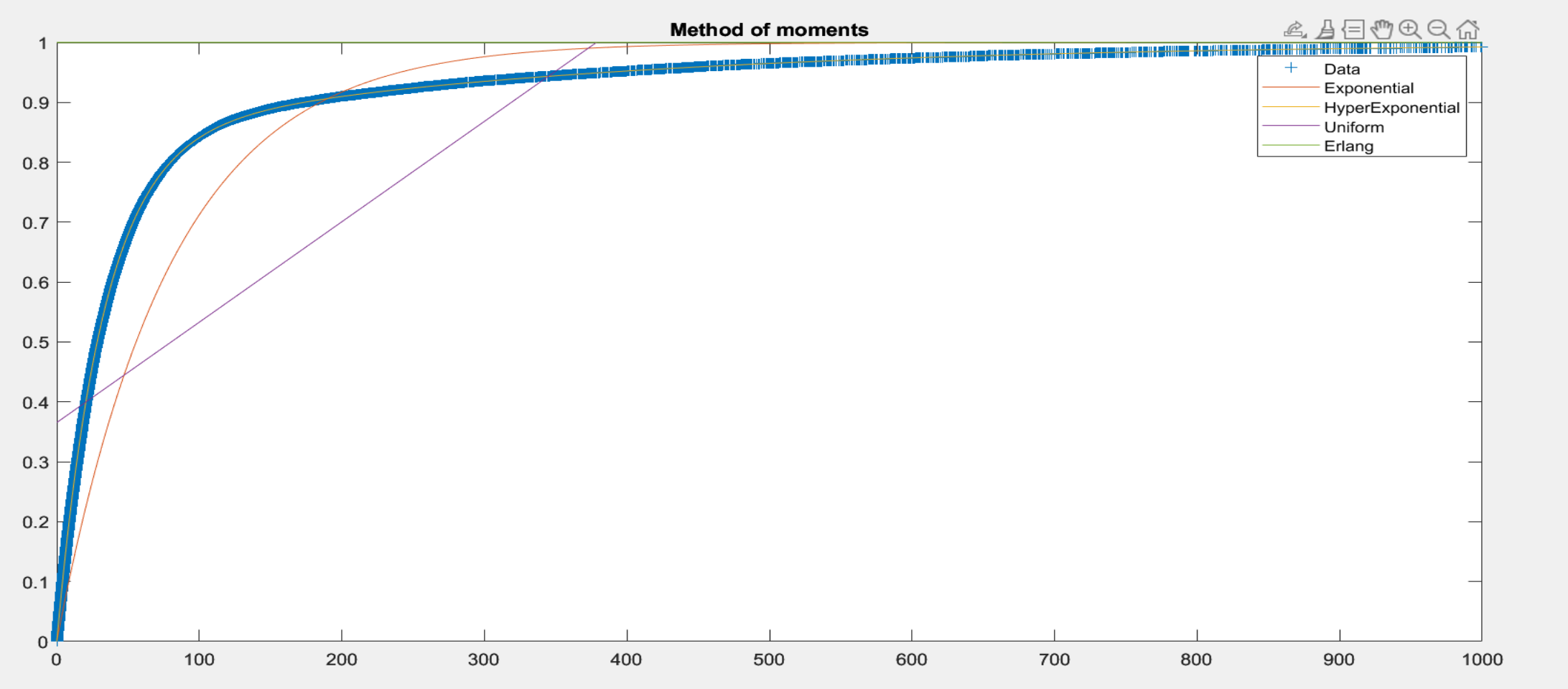
Here the data distribution is perfectly fitted by an Erlang distribution with k = 6 and lambda = 0,0251 .

Fitting – Design Hardware



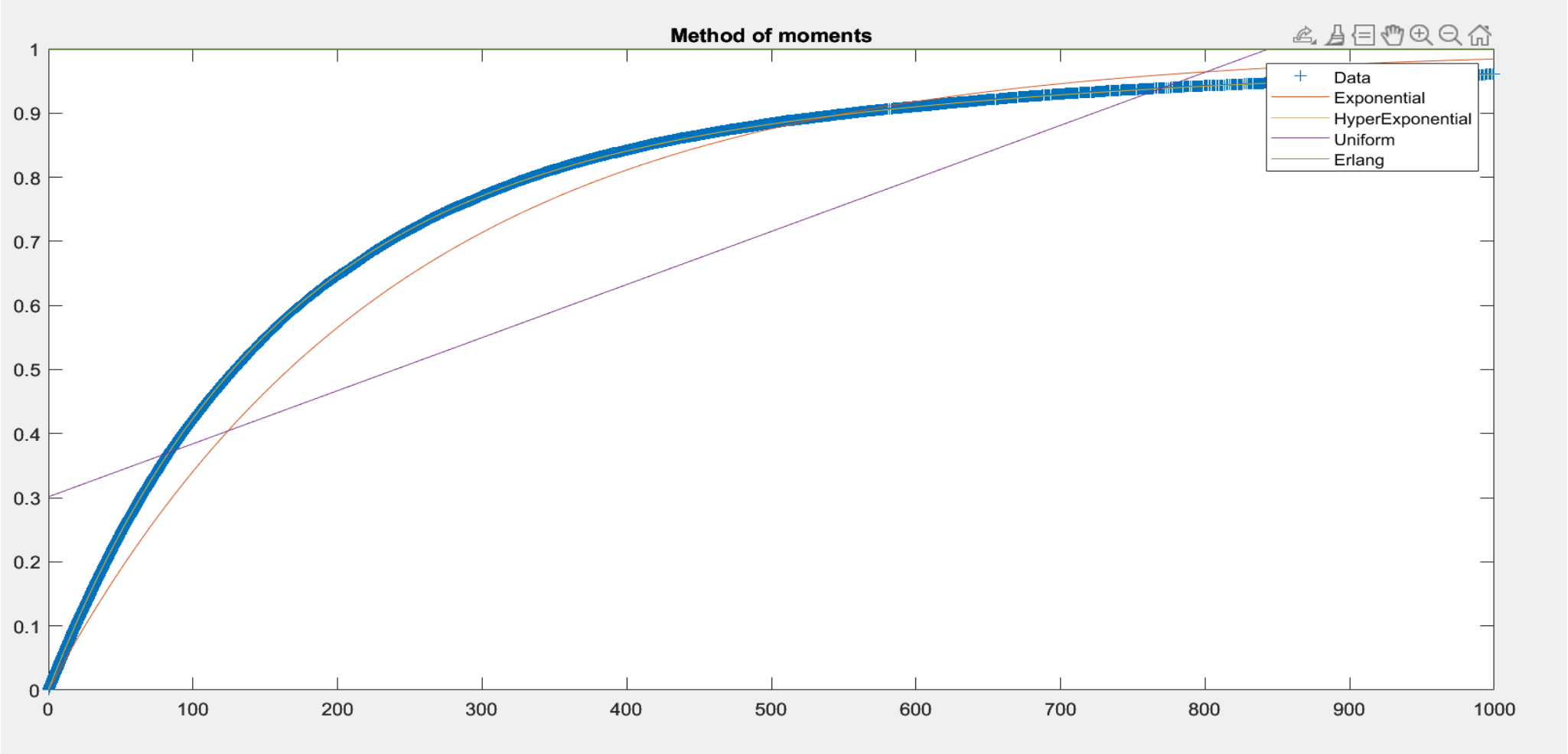
Here the data distribution is perfectly fitted by an Erlang distribution with k = 3 and lambda = 0,0188.

Fitting – Breadboard Hardware



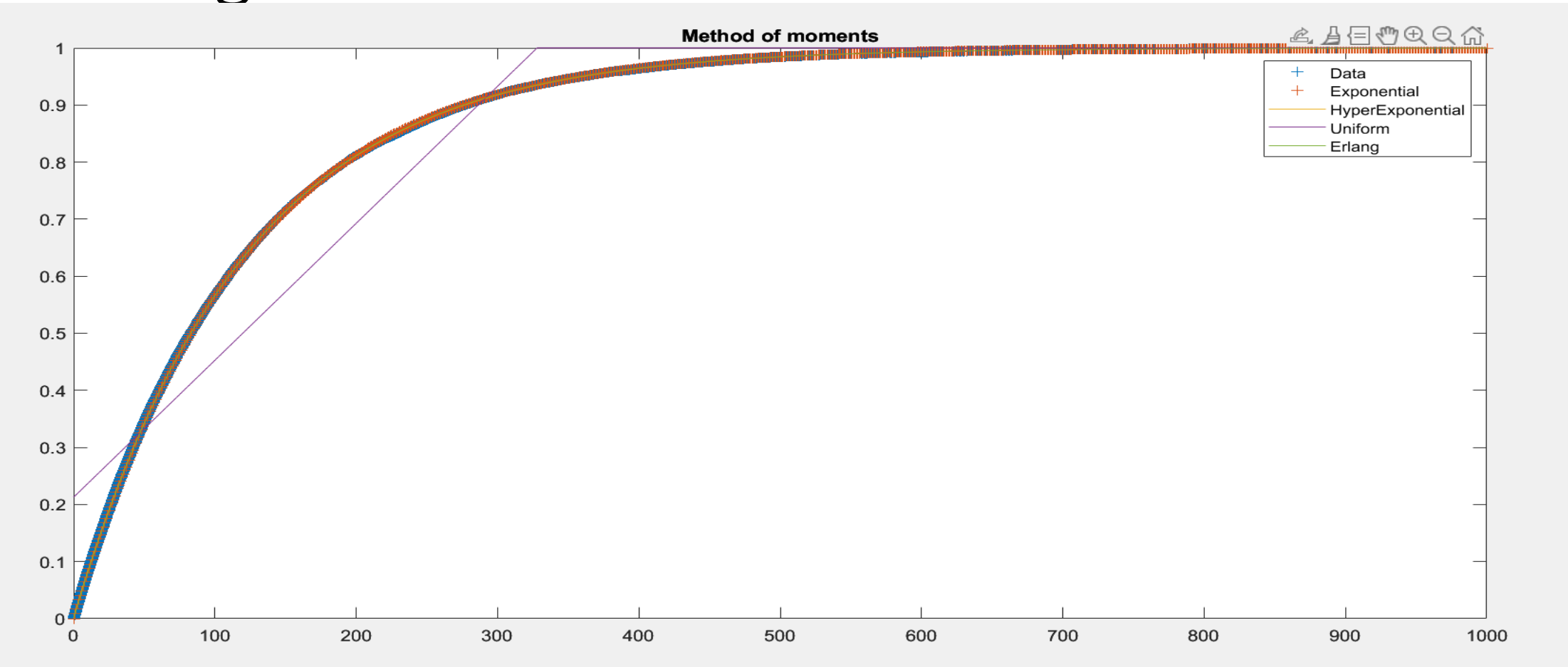
Here the data distribution is perfectly fitted by an HyperExponential distribution with lambda1 =0.003 lambda2=0.03 and p=0.16.

Fitting – Write Software



Here the data distribution is perfectly fitted by an HyperExponential distribution with lambda1 =0.0017 lambda2=0.0068 and p=0.218.

Fitting – Test

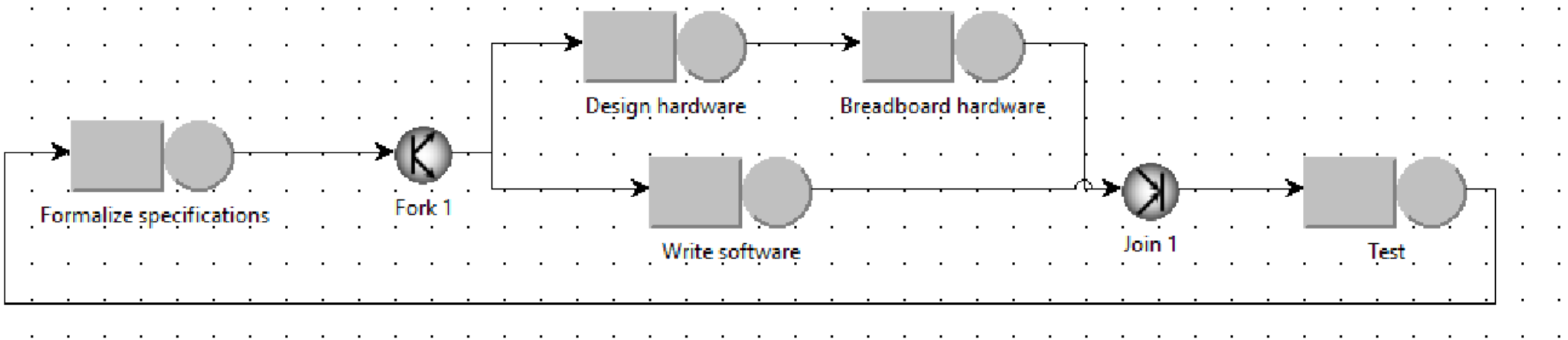


Here the data distribution is perfectly fitted by an HyperExponential distribution with lambda1 =0.008 lambda2=74.54 and p=0.997.

Model Design

To simulate the company's workflow, I implemented a closed model system in JMT and then performed a what-if analysis changing the number of jobs in the system and seeing how the Throughput and the Response Time varied. The JMT model is composed by a queue for each phase of the workflow with the service time distribution as the one fitted in the previously. To simulate the splitting of the jobs in the two parallel branches, I used a fork after the first phase with a corresponding join when the two branches merged. This structure makes it so that a job is splitted after the fork and then whichever part finishes first has to wait for the other branch before going into the test phase.

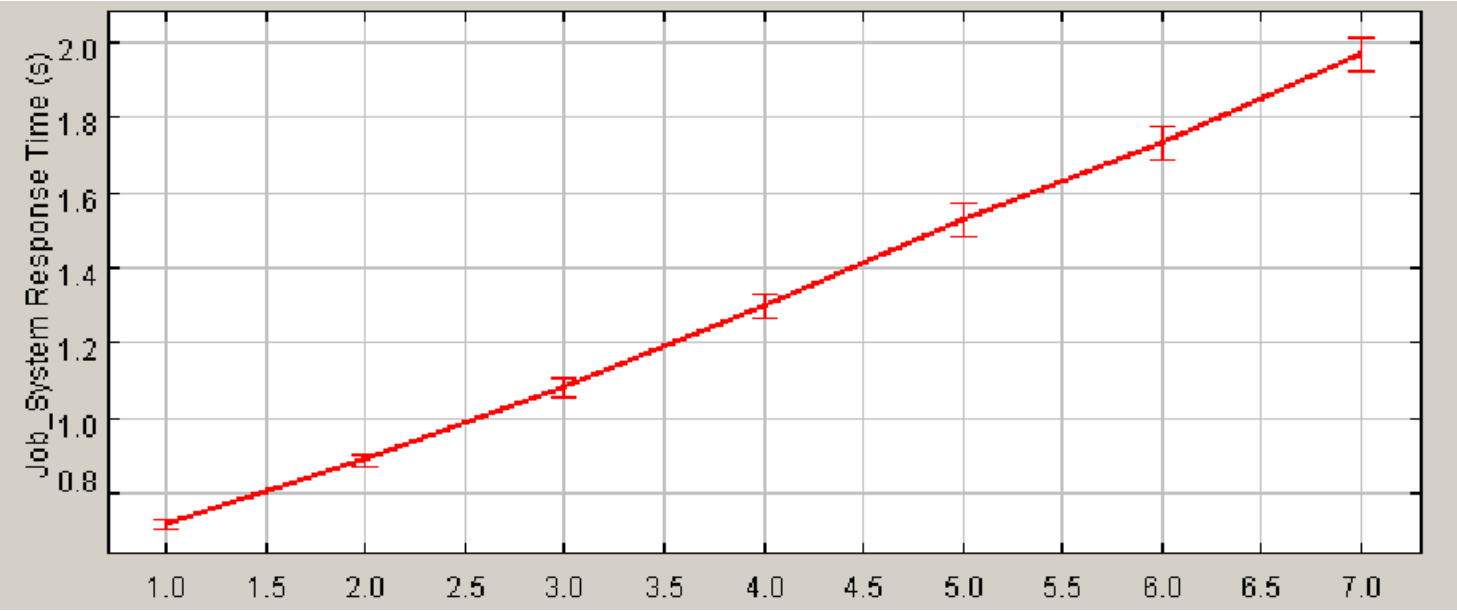
Model Design



Result Analysys

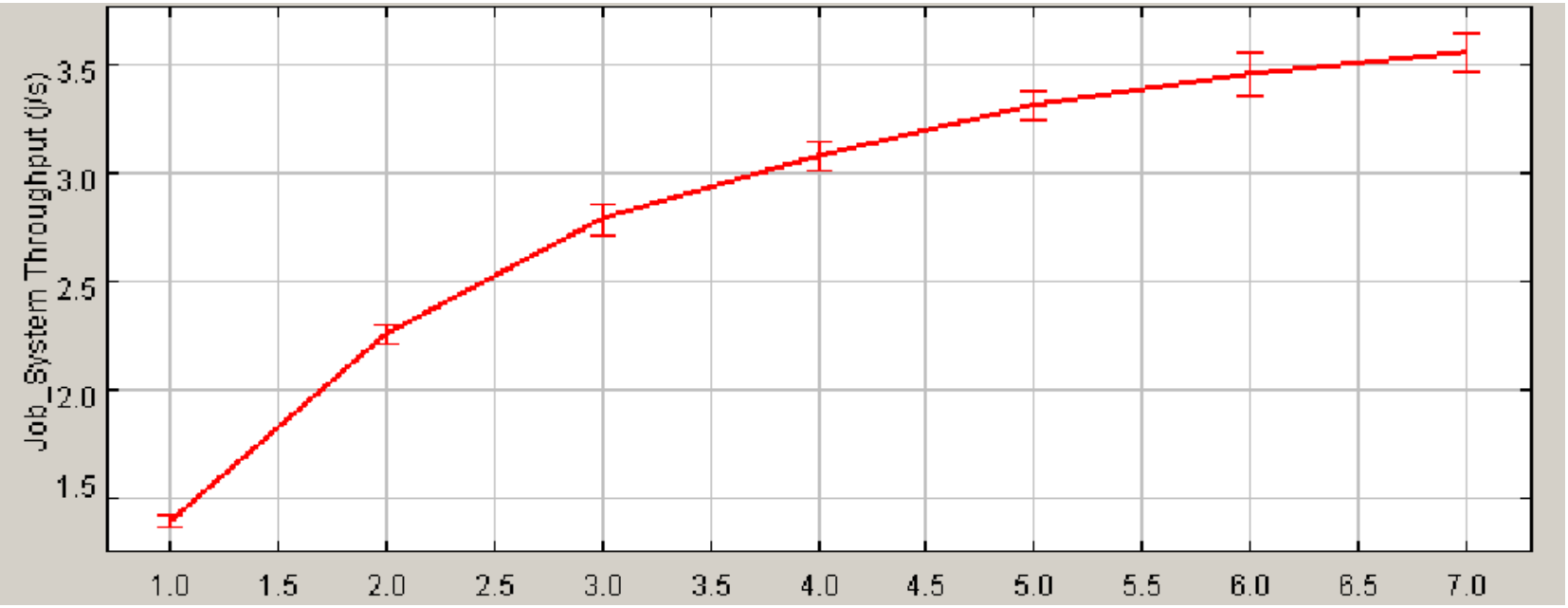
After running the what if analysis for an increasing number of jobs up to 7, I used the results obtained to compute the fraction between the Throughput and the Response Time to see where the balance between the two measures is the best.
In the following slides i will show the results for both the single measures and their fraction. Both the values for the Response Time and for the Throughput were computed with a Confidence Interval of 0.999 and a Maximum Relative Error of 0.03.

Result Analys – Response Time



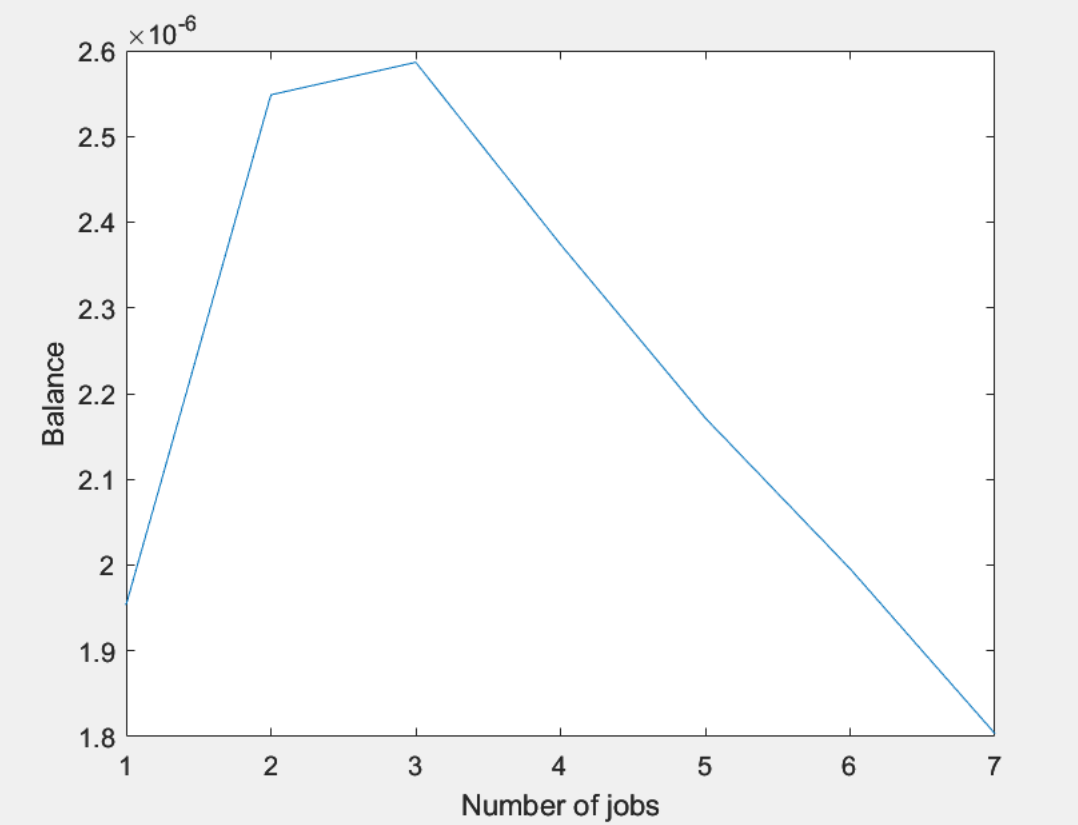
The Response Time (computed in hours) is linearly increasing with number of jobs currently in parallel in the system.

Result Analys – Throughput



The Throughput (computed in jobs/hour) increases until around 5 parallel jobs in the system and then reaches a plateau.

Result Analys – Balance



The Balance(jobs/hour^2) increases with a peak at 3 parallel jobs and then quickly decreases.

Conclusion

As shown by the last graph, we can conclude that the optimal number of jobs for the company to work on in parallel is 3. After this number, the response time (which is increasing linearly) becomes too high while the throughput starts reaching a plateau and slows the improvement.