Distributed Systems, Lab 3 Report

Mislav Jakšić

December 28, 2018

Contents

1 Logical and physical application structure

The application (1) is already abstract and represented as a directed graph of queues.

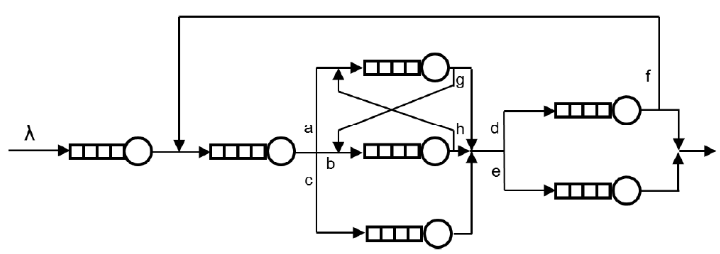


Figure 1: Task 2.2

2 Queueing theory model

Queueing theory, which says that most complex system can be represented as a directed graph of queues, has been applied to the application (1).

A rule of thumb says that in order to calculate the response time T, you should do the following:

1. Find λ_N for every node by solving the matrix and expressing the result as system λ.

2. Find ρ_N for every node using $\rho_N = S_N * \lambda_N$.

3. Find N_N for every node using $N_N = (1 - \rho_N)/\rho_N$.

4. Finally, find T using $T = (N_1 + .. + N_N)/\lambda$.

The analytical solution to is presented in an Microsoft Excel/LibreOffice Calc called "RASUS_LAB_3_Task2_2.ods".

W	Wait time
S	Service time
T	Response time / residual time
λ	Incoming request intensity
δ	Outgoing request intensity
ρ	Demand
N	Queue length
v	Number of visits

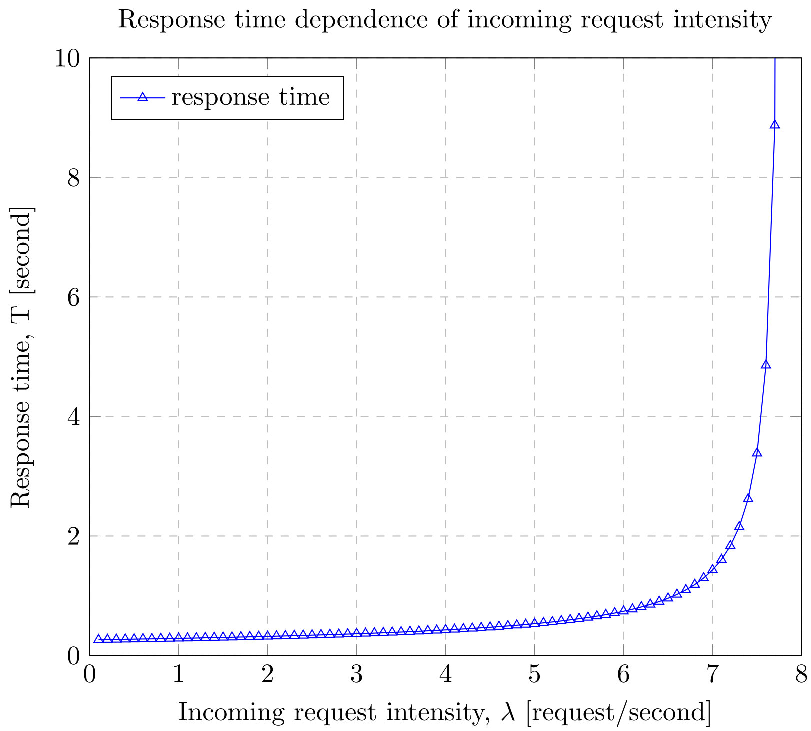
Table 1: Legend

3 Pretty Damn Quick (PDQ) analyzer model

PDQ is a fast queue solver which takes a model as input, written in a number of languages including Perl, Python, C and Java (deprecated).

You can see the PDQ model in the project "Pretty Damn Quick Analysis", "Problem2_2".

4 Results


Response time dependence of incoming request intensity
response time
Response time, T [second]
Incoming request intensity, λ [request/second]
0
2
4
6
8
10
0
1
2
3
4
5
6
7
8