Optimal Scheme for Multi-type Customer Queueing System

Yifan Feng

Department of Mathematics, University of Minnesota

Abstract

In a single server queueing system in which the types of customers are mixed and random like Starbucks, those types of customers who are expected to be slow are likely to block customers behind him, making the overall waiting time unnecessarily long. Hence it is plausible that by splitting the server with properly allocated service resources, each restricting to serve certain types of customers, one can increase the overall service quality.

In this project we built a model describing this phenomenon and discuss the optimal scheme for allocating of service resources. It turns out that there is actually a trade-off between reducing the idle times of both servers and keeping waiting time for the fast-typed customers low. We find that when different types of customers differ significantly in the expected service time, it is good to separate the queue. Moreover, we find some structures in the optimal solution which might reduce the searching complexity into polynomial time.

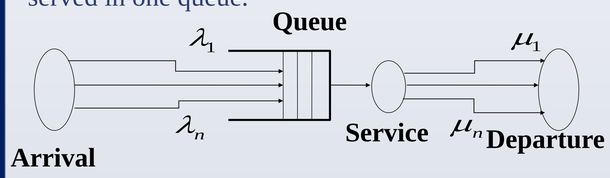
Literature Review

The problem of optimally allocating service resources for queueing system has been studied when the service time of each customer is deterministic, e.g., Mor Harchol-Balte et al (1999, 2002) , Taria et al (2005). However, in real-life service systems like Starbucks, the server is not likely to know the exact serving time for next customer, instead they usually know the distribution of the service time by classifying the demand into different types of drink. This motivates us to divide customers into several types by specifying their arriving rate and expected serving time. As far as we know, this is the first study of such problems under random service time.

Our Model

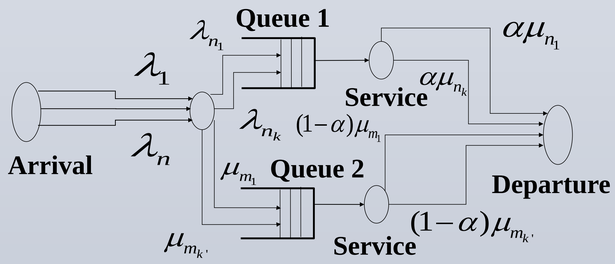
We assume that each type of customer has independent Poisson arrival rate λ_i and Exponential serving rate μ_i We compare the following two schemes.

Under the traditional scheme, all customers are served in one queue.



Scheme 1

Our new scheme is to open up a new server with each focusing on serving fixed types of customers (for simplicity, we only consider two queues in this work). The serving rate of each customer is reduced by a factor which describes the amount of serving resource allocated to each server.



Scheme 2

Let n be the total number of customer types. Let x_i denote whether the ith type of customer is served by the 1st queue. That is, $x_i = 1$ if type i customer is served by the first server, and 0 otherwise. Similarly, let $y_i = 1 - x_i$ denote whether the ith type of customer is served by the 2nd queue. Also let α denote the service capacity allocated to the first server and β denote the service capacity allocated to the second server.

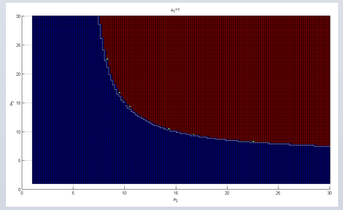
To minimize the average waiting time by each individual, we can formulate our optimization problem as following:

Formulation and Results

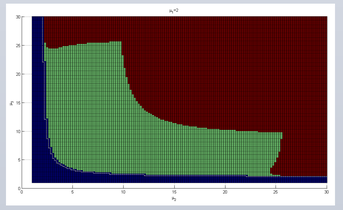
$$
\begin{aligned}
min\quad & \frac{\sum \frac{\lambda_i x_i}{\mu_i^2}}{\alpha^2 - \alpha \sum \frac{\lambda_i x_i}{\mu_i}} \frac{\sum x_i\lambda_i}{\sum \lambda_i} + \frac{\sum \frac{\lambda_i y_i}{\mu_i^2}}{\beta^2 - \beta \sum \frac{\lambda_i y_i}{\mu_i}} \frac{\sum y_i\lambda_i}{\sum \lambda_i} \\
s.t.\quad & x_i + y_i = 1 \\
& \alpha + \beta = 1 \\
& \alpha, \beta, x_i, y_i \geq 0 \\
& x_i, y_i \in \mathbf{Z}
\end{aligned}
$$

When the total number of customer types is reasonably small, we can solve the problem easily by doing line searches for α and enumerate all possible x_i's.

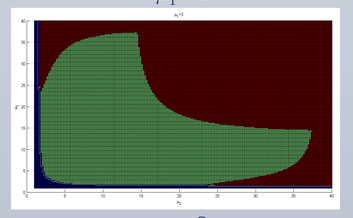
Below is an example when n = 3 and $\lambda_1 = \lambda_2 = \lambda_3 = 1$



$\mu_1 = 1$



$\mu_1 = 2$



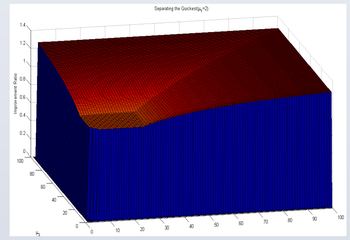
$\mu_1 = 3$

If (μ_1, μ_2, μ_3) lies in the blue area, the queues are not stable; if it lies in the green area, it is better not to separate the queue; if it lies in the red area, queues should be separated.

Results (continued)

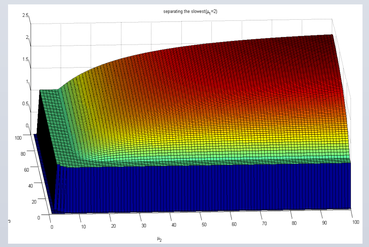
We found that keeping arriving rate identical, if we fix the some serving rates and let the others go to infinity, queues should be separated, as is confirmed by numerical results.

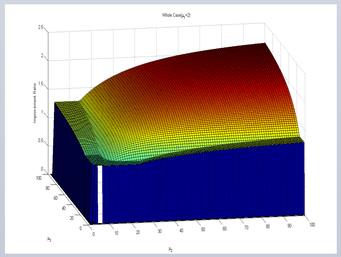
Also, if we look further into the optimal x_i, we found that (1, 0, 1) can not be optimal under any circumstances, as shown by below:



Improvement ratio is defined as the ratio between average waiting time if we don't allow separating queues over that if we do, which is always no less than one.

In the figure above we fix the quickest type of customer to be separated from one server while in the figure left we fix the slowest type to be separated.





By comparing the global optimal case, we found that (1,0,1) never appeared in the optimal solution.

We conjecture that this is true in general, that is the optimal splitting of the queue must be continuous, i.e., there must exist a threshold such that the optimal separation should be made at the threshold. We are still working to prove this conjecture.

Acknowledgements

Please allow me to express my gratitude to my advisor, Prof. Zizhuo Wang for his encouragement all along the way and UROP for the opportunities for such a good hands-on research experience!