Queueing Systems with Abandonment

Study for call centers

Introduction

In a contact center, it is essential to have a robust methodology for estimating the number of agents required each day to maintain a desired service level. The PyWorkforce library offers a function based on Erlang-C models for this purpose. However, the Erlang-C model does not take into account the abandonment rate of incoming calls, which significantly impacts system performance. To address this limitation, Erlang-A models can be utilized, as they incorporate call abandonment as a critical factor in the analysis.
It is worth noting that obtaining the probability of abandonment, a key performance metric, is not straightforward from available literature or online resources regarding Erlang-A models. Different sources provide partial information, and some even require a subscription or purchase of material. This highlights the need for a comprehensive and accessible approach to analyze call center metrics, incorporating both service level targets and abandonment rates.

Methodology

In this study, we will explore three different approaches to analyze call center metrics:
(I) **Exact** resolution, in **call_center_metrics_exact.py**
(II) **Simulation**, in **call_center_metrics_simulation.py**
(III) **Erlang-C** model, in **call_center_metrics_erlangC.py**

While approaches (I) and (II) include abandonment in their calculations, approach (III) is implemented for validation purposes, allowing us to compare the metrics when the abandonment rate is zero. (II) and (III) methods are adaptations from public repositories, see Relevant Links section.

Model Parameters, Variables, and Outputs

GENERAL PARAMETERS

- Lambda (λ) = 120 customers/h (exponential): arrival rate of customers

- Mu (μ) = 4 customers/h (exponential): service rate
- Average Speed of Answer (ASA) = 8 minutes: target waiting time for a transaction in the queue to be attended by a resource

VARIABLES FOR APPROACHES (I) AND (II):
- Abandonment: 0 to 4 calls/h, which means that the average time of abandonment is 15 minutes (4/h)
- Number of agents: ranging from 31 to 35

OUTPUT FOR APPROACHES (I), (II), AND (III):
- Buffer length: average length of the queue
- Waiting probability: probability of a customer waiting in the queue
- Occupancy: proportion of time that an agent is busy
- Abandonment rate: proportion of customers that abandon the call before receiving service

OUTPUT FOR APPROACHES (I) AND (II), AND INPUT VARIABLE FOR APPROACH (III):
- Service level = 0.53 to 0.98: proportion of customers that wait longer than the target waiting time

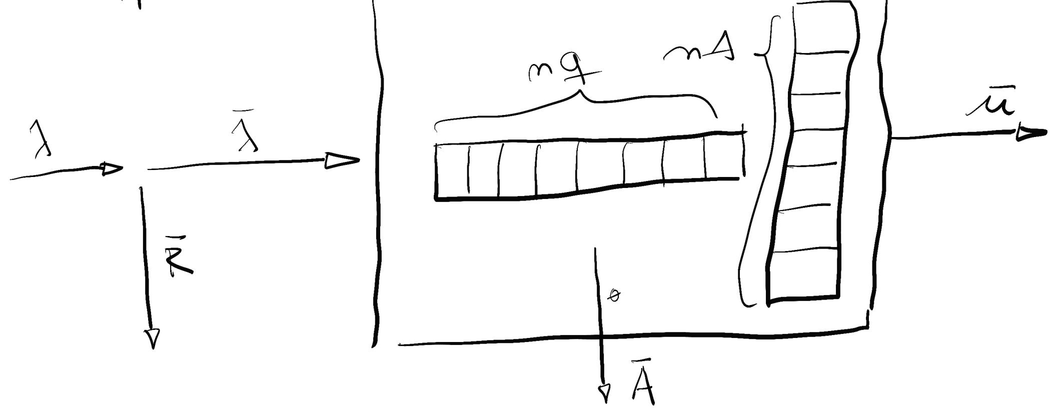
OUTPUT FOR APPROACH (III):
- Number of agents required to achieve the desired service level

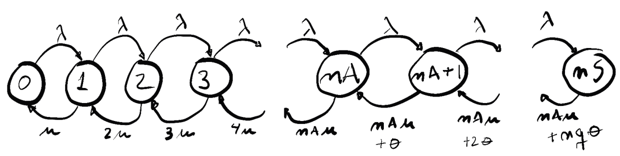
---

Exact Solution

The exact approach for an M/M/k queue system with abandonment is achieved by the resolution of model and the analysis of the behavior of such a system. In an M/M/k queue system, the arrival rate (λ) and service rate (μ) follow exponential distributions, and there are nA servers. Additionally, the system incorporates call abandonment, which is when customers leave the queue before receiving service.

$$nS = nq + nA$$





$$\bar{P} = (P_0, P_1, P_2, P_3 \ldots P_{nA} \ldots P_{nS})$$

$$\bar{P} \cdot \begin{bmatrix} \lambda & 0 & 0 & \cdots & 0 & 0 & 1 \\ \mu & \lambda & 0 & \cdots & 0 & 0 & 1 \\ 0 & 2\mu & \lambda & \cdots & 0 & 0 & 1 \\ \vdots & & & & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & \lambda & 0 & 1 \\ 0 & 0 & 0 & \cdots & nA\mu + (nq-1)\theta & \lambda & 1 \\ 0 & 0 & 0 & \cdots & 0 & nA\mu + nq\theta & 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

To achieve an exact resolution for this queue system, we need to formulate a set of equations that describe the probabilities of various system states. This involves determining the steady-state probabilities for each state (i.e., the number of customers in the system). For an M/M/k queue system with abandonment, the equations can be derived using the balance equations method or the continuous-time Markov chain. The resulting set of equations can be solved using linear algebra. The system, the Markov model, and the equations are shown in the following figure.
The theoretical infinite queue system is truncated at nS (the number of total states) equals to

$$nS = 8 * nA$$

Which means that the system only holds 8 customer per agents in the queue. This does not alter the numerical solution. The equations can be determined by

$$p(n) = \frac{\lambda_{n-1}}{\mu_n} p(n-1)$$

$\lambda_n = \lambda$ as balking is negligible

$$\mu_n = \begin{cases} 0 & \text{for } n = 0 \\ n\mu & \text{for } 1 < n < n_A \\ nA\mu + (n - nA)\theta & \text{for nA <= n <= nS} \end{cases}$$

Once the steady-state probabilities are obtained, key performance metrics can be calculated. An exact resolution provides a precise understanding of the queue system's behavior, helping decision-makers optimize the number of agents and improve the overall performance of the contact center.

Results

The results section of this report is divided into two parts: validation and abandonment analysis.

Validation

In the validation process, we compare the three models (i) exact resolution, (ii) simulation, and (iii) Erlang-C model. The results obtained for Abandon Rate = 0 are exactly the same across all three models, demonstrating their consistency when abandonment is not a factor. This can be verified in the table below, which is also available in the "Erlang" sheet of the "summary_metrics.xlsx" file.

service_level_req	raw_positions	positions	occupancy	waiting_probability	service_level	occ difference w/exact	wait_prob difference w/exact	service_lev diff w/exact	occ difference w/sim	wait_prob difference w/sim	service_lev diff w/sim
0.53	31	31	0.96774	0.79895	0.53130	0.0%	0.0%	0.0%	0.0%	0.2%	-0.2%
0.58	32	32	0.93750	0.63022	0.78311	0.0%	0.0%	0.0%	0.0%	-0.1%	0.0%
0.63	32	32	0.93750	0.63022	0.78311	0.0%	0.0%	0.0%	0.0%	-0.1%	0.0%
0.68	32	32	0.93750	0.63022	0.78311	0.0%	0.0%	0.0%	0.0%	-0.1%	0.0%
0.73	32	32	0.93750	0.63022	0.78311	0.0%	0.0%	0.0%	0.0%	-0.1%	0.0%
0.78	32	32	0.93750	0.63022	0.78311	0.0%	0.0%	0.0%	0.0%	-0.1%	0.0%
0.83	33	33	0.90909	0.49049	0.90097	0.0%	0.0%	0.0%	-0.1%	-1.2%	0.1%
0.88	33	33	0.90909	0.49049	0.90097	0.0%	0.0%	0.0%	-0.1%	-1.2%	0.1%
0.93	34	34	0.88235	0.37638	0.95542	0.0%	0.0%	0.0%	0.0%	0.2%	0.0%

Abandonment Analysis

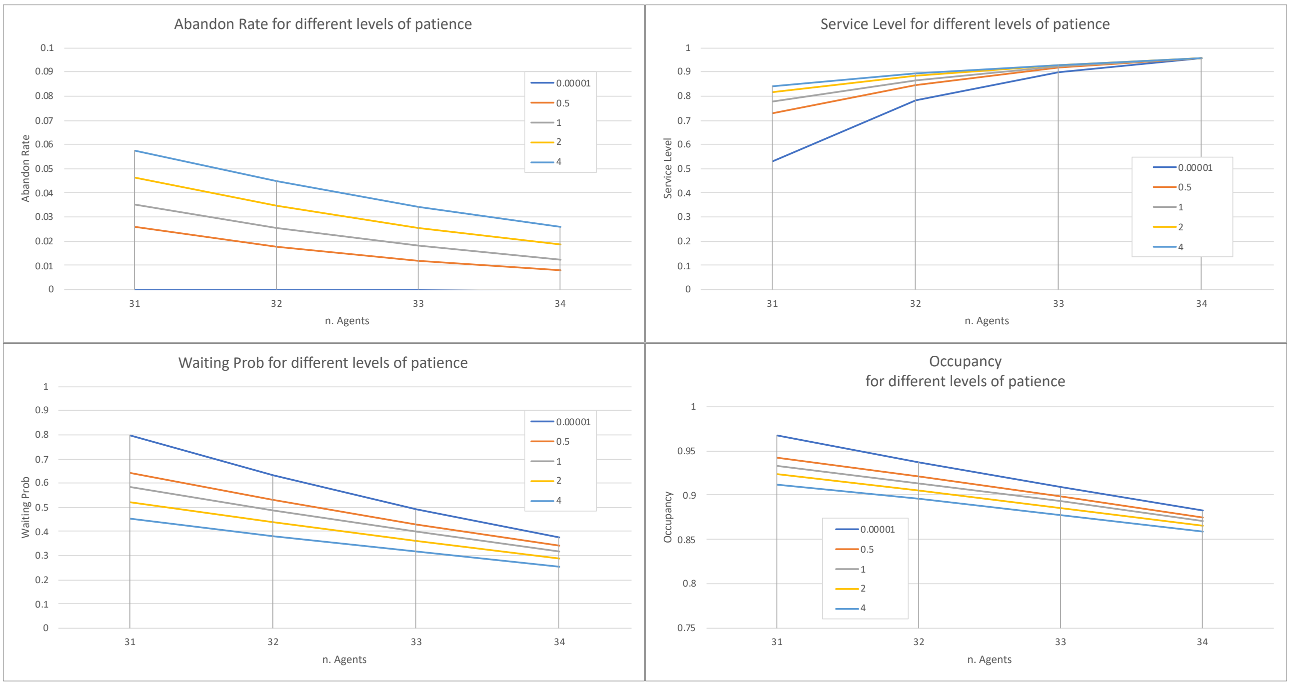
In the abandonment analysis, we present four charts to illustrate various aspects of the system performance, considering the Exact model:

Abandon Rate: This chart analyzes the probability of a call being abandoned by a customer as a function of the number of agents and the patience of the customers. The chart shows the relationship between these variables and the likelihood of call abandonment.

Service Level: In this chart, we utilize the Erlang-C formula for service level while adjusting lambda by the abandon rate and multiplying the service level by (1-AbandonRate). This adapted formula may need further validation and review. The chart displays the service level as a function of the number of agents and the abandonment rate.

Waiting Probability: This chart depicts the probability of a customer having to wait in the queue. It helps visualize how the waiting probability varies with the number of agents and the abandonment rate.

Occupancy: The final chart represents the occupancy, which is the proportion of time that an agent is busy. This chart helps to understand how efficiently the agents are utilized based on the number of agents and the abandonment rate.



By examining these charts, decision-makers can gain insights into the performance of the contact center, allowing them to make informed decisions about staffing and resource allocation to optimize service levels and minimize abandonment rates.

Conclusion

While our models have shed light on various scenarios in a contact center, they need validation with actual data to deliver a more precise understanding of real-world operations.
Consider a case with 32 agents and an average abandonment time of 15 minutes (patience level of 4). Here, the predicted abandonment rate is 4.49%. If the number of agents is increased to 34, the abandonment rate drops by 1.90% to 2.58%.
Assuming an average ticket value of $1000, a recall rate of 60%, and an Arrival Rate of 120 calls per hour, this decrease in abandonment rate substantially impacts revenue. Specifically, 2 more agents will increase revenue by about

$$120 \text{ cust/h} \times 0.019 \times (1 - 0.6) \times 1000 \text{ \$/cust} = 912 \text{ \$/h}$$

This increase in $912 per hour, leads to nearly $7,300 per day or $175,000 per month.

These figures should be considered when deciding on the number of agents. More agents could lead to significant financial returns, improving the contact center's performance and profitability. However, to fully leverage the potential of these models, they need to be calibrated with real data and tested under various operating conditions.

Relevant links

https://towardsdatascience.com/workforce-planning-optimization-using-python-69af0ef9011a
Erlang-C python model and pyworkforce package.

https://www.nextiva.com/blog/call-center-metrics.html
Top 33 call center metrics

https://github.com/yinchi/simpy-examples/blob/main/ex03_mmk_abandonment.py
Script adapted for the Simulation approach.