Introduction to Queueing Theory and Queueing Models

Anthony Hung

2019-03-01

Prerequisites

This vignette assumes a basic understanding of continuous time Markov chains, including birth/death processes.

Introduction

Queueing theory is the study of queues, or waiting lines, which arise in any system where there are demands for a limited amount of resources. Queueing theory involves building models to characterize and evaluate the performance such systems. Examples of queues can be found in broad areas of study from biology (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3737734/), economics (https://www.jstor.org/stable/1832136?seq=1#metadata_info_tab_contents), and healthcare (https://www.sciencedirect.com/science/article/pii/S2211692314000022). By modeling these systems, one can determine adjustments to individual queue parameters that can improve the performance of the queue, which is a central goal in operations management.

What are the components of a queueing process?

In queueing theory terminology, the entities entering and being serviced by a queue are called "customers," and the customers are serviced by "servers." Queues are made up of six fundamental components.

1. The arrival process of customers to the queue
2. The service (departure) time distribution
3. The number of servers in the system
4. The system capacity
5. The total size of calling population
6. The service discipline

Arrival Process

The arrival process describes how customers arrive to the queue. Customers can arrive as individuals or as groups or batches, and arrivals are distributed in time. Arrivals can either occur at regular intervals (deterministic arrival process), at random intervals (e.g. Poisson arrivals), or at time-dependent intervals (time inhomogeneous). Most commonly, the arrival process is modeled as a Poisson process with a single parameter λ, the mean arrival rate.

Service Time Distribution

The service time distribution describes how long the service takes. Like the arrival process, it can be a regular interval (deterministic) or random (e.g. exponentially distributed). Most commonly, the service time distribution is modeled as an exponential distribution. Modeling the service time distribution as exponential essentially means that service is a Poisson process with parameter μ, or the mean service rate.

Server Number

The number of servers that serve customers in the queueing system. Each server has the same service time distribution.

System Capacity

The system capacity describes how many customers can be in the queueing system at once, including customers at servers and customers in the waiting buffer. If the number of customers in the system equals the system capacity, additional customers who arrive at the queue will be “blocked” from joining the queue. In effect, the system capacity places a restriction on the size of the queue no matter how many arrivals occur.

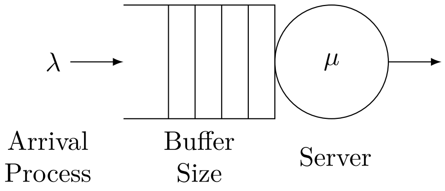
Total Calling Population

The total calling population describes the total number of customers that feed into the queueing system. The number of total customers in a queue system can never be greater than the size of the total calling population, no matter the system capacity.

Service Discipline

The service discipline describes how the order in which customers are served in the queue. The most common and familiar discipline is First in, First out (FIFO), in which customers that have been waiting the longest in the queue are the first to be served. Last in, First out (LIFO), in which customers that have been waiting the shortest are served first, is also known as a stack (as items most recently placed on a stack are on top and therefore also the first to be removed).

A single queue node can be graphically depicted as such:



In this queue, λ describes the arrival process and μ describes the service duration. There is a buffer size of 5, meaning at most 6 customers can be in the system before the next customer is blocked from entereing the queue.

Kendall’s notation for representing queues

English statistician David George Kendall proposed a system for representing properties of queues in a simple way in 1953, and since then Kendall’s notation has been built upon and is widely used in queueing theory today. At it’s core, Kendall’s notation represents each the 6 components of queues: (1) the arrival process A, (2) the service time distribution S, (3) the number of servers c, (4) the number of places in the system K, (5) the calling population N, and (6) the service discipline D through a A/S/c/K/N/D notation containing one coding position representing each component. All queue systems in this notation are assumed to have a single queue no matter the number of servers, and customers move from this single queue to available servers.

Each of the 6 positions in the notation can take on one of several different codes depending on the characteristics of the queue. Some common codes for each component are listed below:

<u>A</u>

Code	Name	Description
M	Markovian or Memoryless	Poisson arrival process
D	Degenerate Distribution	Deterministic or fixed inter-arrival times
G	General Distribution	Inter-arrival times follow an arbitrary distribution

S

Code	Name	Description
M	Markovian or Memoryless	Exponential service times
D	Degenerate Distribution	Deterministic or fixed service times
G	General Distribution	Service times follow an arbitrary distribution

c

Code	Name	Description
1	One Server in the queue	The server has a service time described by S
2	Two Servers in the queue	Each server has a service time described by S

K

Code	Name	Description
∞	∞ buffer size in the queue	There is no limit to the number of customers in the system.

N

Code	Name	Description
∞	∞ customer calling population	Population is unlimited.

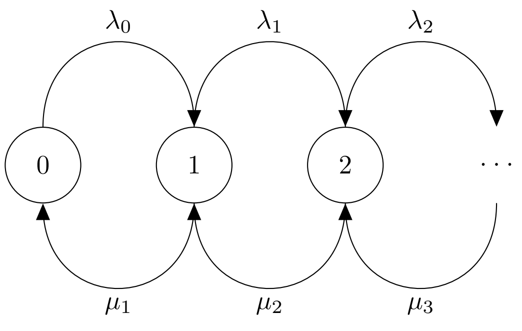
D

Code	Name	Description
FIFO	First in, First out	Customers are served in the order they arrived.
LIFO	Last in, First out	Customers are served in the reverse to the order they arrived.
SIRO	Service in Random Order	Customers are served in random order.
PQ	Priority queueing	Customers are assigned priorities and those with higher priority are served first.

In practice, an abbreviated version of Kendall's notation is used to describe queues, which only includes the first 3 components (A/S/c). When Positions K, N, and D are ommitted, they are assumed to be ∞, ∞, and FIFO respectively. For example, a M/M/1 is abbreviated Kendall's notation for a M/M/1/∞/∞/FIFO queue and has a Poisson arrival process, exponentially distributed service times, 1 server, an infinite system capacity, an infinite calling population, and operates under a FIFO discipline.

Connection between queues and birth/death processes (a continuous time Markov chain)

The behavior of a M/M/c/K/∞/FIFO single queue can be summarized as a Birth/Death process. The Birth/Death process models the arrival process of customers into a queueing system (birth), the departure process of customers out of the queueing system (deaths). In this case, the queue has Poisson arrivals, exponentially distributed service time, c servers, K queue positions, an infinite calling population, and operates under a First in, First out discipline. The Birth/Death process depicted here:



is a M/M/c/∞/∞/FIFO queue when the birth/death process is time-homogeneous ($\lambda_0 = \lambda_1 = \lambda_2 = \cdots = \lambda$ and $\mu_1 = \mu_2 = \mu_3 = \cdots = \mu$).

Modeling the number of customers in a variety of the M/M/1 queue as a continuous time Markov chain.

Taking advantage of the above relationship between time-homogeneous CTMCs (continuous time with discrete state space) and the M/M/1 queue, we can model the number of customers in the queueing system over time as a continuous time Markov chain (CTMC).

The M/M/1 queue has Poisson arrivals (exponential inter-arrival times), with parameter λ the mean arrival rate. At a mean rate of λ, the number of customers in the system moves from state i to state i+1. This queue also has exponentially distributed service times, with parameter μ the mean departure rate. At a mean rate of μ, the number of customers in the system moves from state i to state i-1. At time $t = 0$, the number of customers in the queue $N(t = 0) = 0$.

As the M/M/1 queue has an infinite system capacity and infinite calling population, the state space of the CTMC is $\{0, 1, 2, 3, ...\}$

With the arrival and service processes described above, we can define the transition rate matrix (generator matrix) of the CTMC.

As a reminder, q_{ij} denotes the rate of departing from state i and entering state j for $i \neq j$. $q_{ii} = -\sum\limits_{j \neq i} q_{ij}$ to make the rows of the matrix sum to 0.

$$Q = \begin{bmatrix} -\lambda & \lambda & 0 & 0 & 0 & \dots \\ \mu & -(\mu+\lambda) & \lambda & 0 & 0 & \dots \\ 0 & \mu & -(\mu+\lambda) & \lambda & 0 & \dots \\ 0 & 0 & \mu & -(\mu+\lambda) & \lambda & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

Stationary distribution

Using the transition rate matrix, we can calculate the stationary distribution (π) of the system. *As a note,* in order to reach a steady state, $\mu > \lambda$. Otherwise, the system will never reach a steady state and customers will continually be added to the queue at a rate higher than customers are serviced ($\mu < \lambda$), or the system will be a random walk ($\mu = \lambda$). The stationary distribution describes the proportion of time that the system has n customers when the system is at a steady state.

We can use the local balance equation $\pi_i q_{ij} = \pi_j q_{ji}$ to find the stationary distribution.

$$\pi_0 q_{01} = \pi_1 q_{10} \longrightarrow \pi_0 \lambda = \pi_1 \mu$$

$$\pi_1 q_{12} = \pi_2 q_{21} \longrightarrow \pi_1 \lambda = \pi_2 \mu$$

$$\pi_2 q_{23} = \pi_3 q_{32} \longrightarrow \pi_2 \lambda = \pi_3 \mu$$

$$\pi_1 = \pi_0 \frac{\lambda}{\mu}$$

$$\pi_2 = \pi_1 \frac{\lambda}{\mu} = \pi_0 \frac{\lambda^2}{\mu^2}$$

Generally,

$$\pi_n = \pi_0(\frac{\lambda}{\mu})^n$$

Because $\sum_{n=0}^{\infty} \pi_n = 1$, $\pi_0 = \frac{1}{1+\sum_{n=1}^{\infty}(\frac{\lambda}{\mu})^n}$. Because $\lambda < \mu$, the term $\sum_{n=1}^{\infty}(\frac{\lambda}{\mu})^n$ is a geometric series and $\sum_{n=1}^{\infty}(\frac{\lambda}{\mu})^n = \frac{\frac{\lambda}{\mu}}{1-\frac{\lambda}{\mu}}$. Substituting the term back in to the overall equation, $\pi_0 = \frac{1}{1+\frac{\frac{\lambda}{\mu}}{1-\frac{\lambda}{\mu}}} = 1 - \frac{\lambda}{\mu}$

Therefore, the general solution is:

$$\pi_n = \pi_0(\frac{\lambda}{\mu})^n = (1 - \frac{\lambda}{\mu})(\frac{\lambda}{\mu})^n$$

$$

In the next vignette, we will look at examples of simulating and measuring the performance of queues.