CS 547 Lecture 36: Multi-Server Queues

Daniel Myers

M/M/∞

Consider a queue that has an infinite number of servers, so that every customer that arrives can immediately enter service, and there is never anyone waiting. We've used the M/M/∞ queue as an element in most of our models, but we've usually treated it as a *delay* node rather than an actual queueing model.

We can use a Markov chain to solve for the queue length distribution of the M/M/∞ system. Let the Poisson arrival rate be λ and let each of the infinite number of servers have exponential service rate μ.

When there are k customers in the system, there are k servers working in parallel. We previously established that the *minimum* of k iid exponential random variables was also exponentially distributed with parameter $k\mu$. Therefore, the overall completion rate when there are k customers in the M/M/∞ system is $k\mu$.

We'll solve the Markov chain model by setting up and solving the balance equations for each state until we notice a clear pattern, then use total probability to solve for the initial condition π_0.

First, consider state 0. The rate of leaving state 0 due to arrivals is $\pi_0\lambda$. The rate of entering state 0 due to departures from state 1 is $\pi_1\mu$.

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

Now analyze state 1. The total rate of leaving state 1 is $(\lambda+\mu)\pi_1$. The rate of entering state 1 due to arrivals from state 0 is $\lambda\pi_0$ and the rate of entering due to departures from state 2 is $2\mu\pi_2$, where the 2 is a consequence of having two customers in service simultaneously, as discussed above.

$$(\lambda+\mu)\pi_1 = \lambda\pi_0 + 2\mu\pi_2$$

Solving for π_2 in terms of π_0,

$$\pi_2 = \frac{1}{2}\left(\frac{\lambda}{\mu}\right)^2 \pi_0$$

The balance equation for state 2 is

$$(\lambda+2\mu)\pi_2 = \lambda\pi_1 + 3\mu\pi_3$$

Solving for π_3,

$$\pi_3 = \frac{1}{3\cdot 2}\left(\frac{\lambda}{\mu}\right)^3 \pi_0$$

Solving a few more equations will reveal the final pattern.

$$\pi_k = \frac{1}{k!}\left(\frac{\lambda}{\mu}\right)^k \pi_0$$

Using total probability to solve for π_0,

$$\sum_{k=0}^{\infty} \frac{1}{k!} \left(\frac{\lambda}{\mu}\right)^k \pi_0 = 1$$

The summation simplifies to $e^{\frac{\lambda}{\mu}}$. Solving for π_0,

$$\pi_0 = e^{-\frac{\lambda}{\mu}}$$

The final formula for π_k shows that the number of customers in the M/M/∞ queue has a Poisson distribution with parameter $\frac{\lambda}{\mu}$.

$$\pi_k = \frac{1}{k!} \left(\frac{\lambda}{\mu}\right)^k e^{-\frac{\lambda}{\mu}}$$

M/M/m

Now consider a multi-server queue with m identical servers, each operating at rate μ. Customers that arrive when a server is free can enter service immediately; if all servers are occupied, customers will wait in FCFS order until someone departs and a server becomes available.

The model has two basic cases. When there are $k \leq m$ customers in service, the overall completion rate is $k\mu$, just like the M/M/∞ model. When there are $k \geq m$ customers in the model, all of the servers are occupied, and the completion rate is fixed at $m\mu$.

For state $k < m$, the solution to the balance equation is the same as M/M/∞,

$$\pi_k = \frac{1}{k!} \left(\frac{\lambda}{\mu}\right)^k \pi_0$$

When $k \geq m$, the solution changes to incorporate powers of $\frac{1}{m}$.

$$\pi_k = \left(\frac{1}{m}\right)^{k-m} \frac{1}{m!} \left(\frac{\lambda}{\mu}\right)^k \pi_0$$

We can add both cases together to obtain one expression for π_0.

$$\sum_{k=0}^{m-1} \frac{1}{k!} \left(\frac{\lambda}{\mu}\right)^k \pi_0 + \sum_{k=m}^{\infty} \left(\frac{1}{m}\right)^{k-m} \frac{1}{m!} \left(\frac{\lambda}{\mu}\right)^k \pi_0 = 1$$

The final expression is not pretty, but uses only basic parameters and a finite sum, so it can still be calculated efficiently

$$\pi_0 = \left(\sum_{k=0}^{m-1} \frac{1}{k!} \left(\frac{\lambda}{\mu}\right)^k + \frac{\left(\frac{\lambda}{\mu}\right)^m}{m!(1 - \frac{\lambda}{m\mu})} \right)^{-1}$$

One other interesting multi-server model is the M/M/m/m queue, which has m servers, but no waiting space. Customers that arrive when all servers are busy are dropped and lost. This was one of the original queueing models studied by Erlang in his analysis of telephone networks.