CS 547 Lecture 14: Other Service Time Distributions

Daniel Myers

Deterministic

The M/D/1 queue has a *deterministic* service time distribution, where all customers receive exactly the same service, s. There is no variability in the service times, so $c_s^2 = 0$.

We can use the tagged customer method to discover the average residence time in the M/D/1 queue. Recall the master tagged customer equation.

$$\overline{R} = U\overline{z} + (\overline{Q} - U)\overline{s} + \overline{s}$$

where $\overline{z}$ represents the expected residual life.

In the M/M/1 queue, we were able to use the memoryless property of the exponential distribution to argue that $\overline{z} = \overline{s}$. Deterministic service times are not memoryless, so we need to derive the correct value for $\overline{z}$ to solve the equation for $\overline{R}$.

All customers receive a service length of exactly s. By the PASTA property, a newly arriving customer arrives at a random moment in time, which is like arriving at a random moment during the service period of length s. On average, therefore, a newly arriving customer arrives halfway through the service period and waits for an average of $\frac{s}{2}$.

Substituting this value and solving for $\overline{R}$ yields

$$\overline{R} = \frac{s(1 - \frac{U}{2})}{1 - U}$$

When U approaches 1, the M/D/1 residence time is close to half that of the M/M/1 queue.

This example illustrates an important result: if we know the average residual service time for a particular service distribution, we can use it to derive the average residence time.

Erlang-k

We encountered the Erlang distribution in our derivation of the M/M/1 residence time distribution. Conceptually, an Erlang-k distribution consists of k service stages, each exponentially distributed with the same rate parameter μ. A customer advances through the stages, receiving exponentially distributed service at each one. A customer's service is complete when it finishes all k stages.

Note that the stages are only a conceptual model for the service behavior – the queue still has only one server and only serves one customer at a time.

We can derive the Erlang distribution by reasoning about the behavior of a customer as it moves through the stages. A customer will only be in service at time t if it has completed *fewer* than k stages by time t.

The time between stage completions is exponentially distributed, so the Poisson distribution describes the probability of completing a given number of stages by time t. The CDF is thus

$$F(t) = P[X \leq t] = 1 - \sum_{j=0}^{k-1} \frac{e^{-\mu t}(\mu t)^j}{j!}$$

The summation uses the Poisson distribution to calculate the total probability of getting *fewer* than k stage completions by time t.

The Erlang-k distribution has $\overline{s} = \frac{k}{\mu}$ and $c_s^2 = \frac{1}{k}$. Intuitively, as the number of stages increases, the variability of each individual stage is smoothed out and balanced by the other stages.

Pareto

The Pareto distribution is the classic heavy-tailed distribution. In comparison with the exponential, it has a much higher probability of generating extreme values. This means that jobs with very long service times account for a significant fraction of the queue's total work.

The Pareto distribution is often associated with the famous *80-20 rule*, which holds that 80% of outputs are attributable to only 20% of inputs in applications with heavy-tailed behavior. For example, it's been observed that 20% of a population tends to hold about 80% of total wealth, or that 80% of business sales revenue tends to come from only 20% of customers. An extension of this rule holds that the top 1% of inputs account for 50% of outputs. If a system's jobs are Pareto distributed, then half of the total system running time will be dedicated to serving only 1% of jobs!

It's important to remember that the numbers 80 and 20 are not magical. The actual values will vary for different applications. They don't even need to sum to one, since they're measures of two different quantities. The significant part of the "law of the vital few," as it's sometimes called, is the relative importance of a surprisingly small portion of the population.

The CDF of the Pareto distribution is

$$F(x) = 1 - \left(\frac{k}{x}\right)^{\alpha}$$

Here, k is the distibution's minimum value and α is the scale parameter.

When $\alpha > 1$, the mean of the Pareto distribution is

$$\overline{s} = \frac{\alpha k}{\alpha - 1}$$

If $\alpha \leq 1$ the mean is ∞ – the distribution is so biased towards extreme values that it's impossible to calculate an average!

Hyperexponential

The hyperexponential distribution is formed from a probabilistic mixture of exponentials. A *two-stage hyperexponential* is formed by a mixture of two exponentials and has three parameters: p, μ_1, and μ_2. A customer receives exponentially distributed service at rate μ_1 with probability p and at rate μ_2 with probability $1 - p$.

It's possible to define hyperexponentials with any number of stages, provided you specify enough parameters to calculate probabilities of receiving service from each stage and the rate of service in each stage.

The hyperexponential distribution is often used to model applications where $c_s^2 > 1$, as it is more analytically tractable than the Pareto or other heavy-tailed distributions. Further, it's possible to calculate parameters for a two-stage hyperexponential distribution that will match any desired mean and variance, making it easy to fit a hyperexponential service time to observed data.

Note that a two-stage hyperexponential is not the same as a queue with two servers! At any time, there is at most one customer being served, but that customer's actual service distribution is chosen from one of the two exponential distributions.