

## 6 M/M/∞ Queuing Systems

Let us now consider an unlimited number of servers  $k = \infty$ . That *completely* eliminates the waiting time. Whenever a job arrives, there will always be servers available to handle it and thus,

$X = X_s$ , the number of jobs in the system is the number of jobs receiving service,

$R = S$ , response time consists of service time only,

$X_w = 0$ , no jobs waiting in queue,

$W = 0$ , no waiting time.

All the formulas we derived for M/M/k systems apply to M/M/∞ systems, by letting the number of servers  $k \rightarrow \infty$ . Let us see what we get.

First off, the number of jobs will always be less than the number of servers ( $i < k$ ), so we always have  $n = i$ . That is, with  $i$  jobs in the system, exactly  $i$  servers are busy.

The transition probability matrix for the number of jobs in the system,  $X$ , is given by

$$P = \begin{bmatrix} 1 - p_A & p_A & 0 & 0 & 0 & 0 & \dots \\ p_S & 1 - p_A - p_S & p_A & 0 & 0 & 0 & \dots \\ 0 & 2p_S & 1 - p_A - 2p_S & p_A & 0 & 0 & \dots \\ 0 & 0 & 3p_S & 1 - p_A - 3p_S & p_A & 0 & \dots \\ 0 & 0 & 0 & 4p_S & 1 - p_A - 4p_S & p_A & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}.$$

Let us see what becomes the steady-state distribution. The first component,  $\pi_0$  becomes

$$\pi_0 = P(X = 0) = \frac{1}{\sum_{i=0}^{\infty} \frac{r^i}{i!} + \lim_{k \rightarrow \infty} \left( \frac{r^k}{k!} \cdot \frac{1}{1 - r/k} \right)}.$$

Now,

$$\lim_{k \rightarrow \infty} \left( \frac{r^k}{k!} \cdot \frac{1}{1 - r/k} \right) = \lim_{k \rightarrow \infty} \frac{r^k}{k!} = 0,$$

because the factorial converges faster to  $\infty$  than the exponential function. The other term is the Taylor series of the function  $e^r$ , so the steady-state distribution is

$$\begin{aligned}\pi_0 &= e^{-r}, \\ \pi_i &= \frac{r^i}{i!} e^{-r}, \quad \forall i \geq 1.\end{aligned}\tag{6.1}$$

So the pdf of  $X(t)$ , the number of concurrent jobs in an  $M/M/\infty$  system at time  $t$ , is

$$X(t) \left( \frac{r^i}{i!} e^{-r} \right)_{i=0,1,\dots},\tag{6.2}$$

a Poisson distribution with parameter  $r = \frac{\lambda_A}{\lambda_S}$  (which can be arbitrarily large), with mean and variance

$$E(X) = V(X) = r.\tag{6.3}$$

**Remark 6.1.** Clearly, nobody can physically build an *infinite* number of devices. In practice, having an unlimited number of servers simply means that any number of concurrent jobs can be served simultaneously. Example: internet service providers or telephone companies (which allow virtually any number of concurrent connections), an unlimited number of people can watch a TV channel or listen to a radio station, etc. A model with infinitely many servers is a reasonable approximation for a system where jobs typically don't wait and get their service immediately. This may be appropriate for a computer server, a grocery store, Facebook, etc.

**Example 6.2.** A certain powerful server can afford practically any number of concurrent users. Users connect to the server at random times, every 3 minutes, on the average, according to a Poisson counting process. Each user spends an Exponential amount of time on the server with an average of 1 hour and disconnects from it, independently of other users. Find

- the fraction of time when no users are connected to the server;
- the expected number of concurrent users at any time;
- if a message is sent to all users, the probability that 15 or more users will receive this message immediately.

**Solution.** This fits the description of an  $M/M/\infty$  system with

$$\begin{aligned}\lambda_A &= 1/3 \text{ / minute,} \\ \mu_S &= 1 \text{ hour} = 60 \text{ minutes, so} \\ \lambda_S &= 1/\mu_S = 1/60 \text{ / minute,} \\ r &= \frac{\lambda_A}{\lambda_S} = \frac{1/3}{1/60} = 20.\end{aligned}$$

The number of concurrent users has  $Poiss(20)$  distribution.

a)

$$P(X = 0) = \pi_0 = e^{-20} = 2.06 \cdot 10^{-9} = 0.$$

This server is practically *never* idle.

b) The expected number of concurrent users is

$$E(X) = r = 20 \text{ users.}$$

Also, if an urgent message is sent to all the users, then 20 users, on the average, will see it immediately.

c) Fifteen or more users will receive a message immediately if 15 or more users are connected, so, with probability

$$\begin{aligned}P(X \geq 15) &= 1 - P(X < 15) = 1 - P(X \leq 14) \\ &= 1 - \text{poisscdf}(14, 20) = 0.8951.\end{aligned}$$

■

## 7 Simulation of Queuing Systems

We developed a theory and understood how to analyze and evaluate rather basic queuing systems: Bernoulli and  $M/M/k$ . Most of the results were obtained from the Markov property of the considered queuing processes. For these systems, we derived a steady-state distribution of the number of concurrent jobs and computed the vital performance characteristics from it.

In practice, however, many queuing systems have a rather complex structure. Jobs may arrive according to a non-Poisson process, often the rate of arrivals changes during the day (there is a

rush hour on highways or on the internet, etc.). Service times may have different distributions and they are not always memoryless, thus the Markov property may not be satisfied. The number of servers may also change during the day (additional servers may turn on during rush hours). Some customers may get dissatisfied with a long waiting time and quit in the middle of their queue. And so on. Queuing theory does not cover all the possible situations. On the other hand, we can simulate the behavior of almost any queuing system and study its properties by Monte Carlo methods.

A queuing system is Markov only when its interarrival and service times are memoryless. Then the future can be predicted from the present without relying on the past. It can be simulated using the algorithm given for Markov chains (Algorithm 2.11 in Lecture 5). To study long-term characteristics of a queuing system, the initial distribution of  $X_0$  typically does not matter, so we may start this algorithm with 0 jobs in the system and then “switch on” the servers.

Even when the system is Markov, some interesting characteristics do not follow from its steady-state distribution directly, but they can be estimated from a Monte Carlo study.

Performance of more complicated and advanced queuing systems can be evaluated by Monte Carlo methods. One needs to simulate arrivals of jobs, assignment of servers and service times and to keep track of all variables of interest. Monte Carlo methods of Chapter 2 let us simulate and evaluate rather complex queuing systems far beyond Bernoulli and  $M/M/k$ . As long as we know the distributions of interarrival and service times, we can generate the processes of arrivals and services. To assign jobs to servers, we keep track of servers that are available each time when a new job arrives. When all the servers are busy, the new job will enter a queue. As we simulate the work of a queuing system, we keep records of events and variables that are of interest to us. After a large number of Monte Carlo runs, we average our records in order to estimate probabilities by long-run proportions and expected values by long-run averages.