Chapter 1: OVERVIEW OF QUEUEING THEORY

1. Probability Theory

In the study of a queueing system, we are presented with a very dynamic picture of events happening within the system in an apparently random fashion. Neither do we have any knowledge about when these events will occur nor are we able to predict their future developments with certainty. Mathematical models have to be built and probability distributions used to quantify certain parameters in order to render the analysis mathematically tractable. The importance of probability theory in queueing analysis cannot be over-emphasized. It plays a central role as that of the limiting concept to calculus. The development of probability theory is closely related to describing randomly occurring events and has its roots in predicting the random outcome of playing games. We shall begin by defining the notion of an event and the sample space of a mathematical experiment which is supposed to mirror a real-life phenomenon.

1.1 Sample Spaces and Axioms of Probability

A *sample space* (Ω) of a random experiment is a collection of all the mutually exclusive and exhaustive simple outcomes of that experiment. A particular simple outcome (ω) of an experiment is often referred to as a *sample point*. An *event* (E) is simply a subset of Ω and it contains a set of sample points that satisfy certain common criteria. For example, an event could be the even numbers in the toss of a dice and it contains those sample points {[2], [4], [6]}. We indicate that the outcome ω is a sample point of an event E by writing $\{\omega \in E\}$. If an event E contains no sample points, then it is a null event and we write $E = \varnothing$. Two events E and F are said to be mutually exclusive if they have no sample points in common, or in other words the intersection of events E and F is a null event, i.e. $E \cap F = \varnothing$.

1.1 Sample Spaces and Axioms of Probability

There are several notions of probability. One of the classic definitions is based on the relative frequency approach in which the probability of an event E is the limiting value of the proportion of times that E was observed. That is

$$P(E) = \lim_{N \to \infty} \frac{N_E}{N} \tag{1.1}$$

where N_E is the number of times event E was observed and N is the total number of observations. Another one is the so-called axiomatic approach where the probability of an event E is taken to be a real-value function defined on the family of events of a sample space and satisfies the following conditions:

1.1 Sample Spaces and Axioms of Probability

Axioms of probability

(i) $0 \le P(E) \le 1$ for any event in that experiment
(ii) $P(\Omega) = 1$
(iii) If E and F are mutually exclusive events, i.e. $E \in F = \varnothing$, then $P(E \cup F) = P(E) + P(F)$

Proposition 1.1

(i) $P(\varnothing) = 0$
(ii) $P(\bar{E}) + P(E) = 1$ for any event E in Ω, where $\bar{E}$ is the compliment of E.
(iii) $P(E \cup F) = P(E) + P(F) - P(E \cap F)$, for any events E and F.
(iv) $P(E) \le P(F)$, if $E \subseteq F$.
(v) $P\left(\bigcup_i E_i\right) = \sum_i P(E_i)$, for $E_i \cap E_j = \varnothing$, when $i \neq j$.

1.1 Sample Spaces and Axioms of Probability

Example 1:

Let us suppose a tourist guide likes to gamble with his passengers as he guides them around the city on a bus. On every trip, there are about 50 random passengers. Each time he challenges his passengers by betting that if there is at least two people on the bus that have the same birthday, then all of them would have to pay him $1 each. However, if there were none for that group on that day, he would repay each of them $1. What is the likelihood (or probability) of the event that he wins his bet?

1.1 Sample Spaces and Axioms of Probability

Solution:

Let us assume that each passenger is equally likely to have their birthday on any day of the year (we will neglect leap years). In order to solve this problem we need to find the probability that nobody on that bus has the same birthday. Imagine that we line up these 50 passengers, and the first passenger has 365 days to choose as his/her birthday. The next passenger has the remainder of 364 days to choose from in order for him/her not to have the same birthday as the first person (i.e. he has a probability of 364/365). This number of choices reduces until the last passenger. Therefore:

$$P(\text{None of the 50 passengers has the same birthday}) = \underbrace{\left(\frac{364}{365}\right)\left(\frac{363}{365}\right)\left(\frac{362}{365}\right)\cdots\left(\frac{365-49}{365}\right)}_{49 terms}$$

1.1 Sample Spaces and Axioms of Probability

Solution:

Therefore, the probability that the tourist guide wins his bet can be obtained by Proposition 1.1 (ii):

$$P(\text{At least 2 passengers out of 50 has the same birthday}) = 1 - \left(\frac{\prod_{j=1}^{49}(365-j)}{365^{49}} \right) = 0.9704.$$

1.2 Conditional Probability and Independence

In many practical situations, we often do not have information about the outcome of an event but rather information about related events. Is it possible to infer the probability of an event using the knowledge that we have about these other events? This leads us to the idea of *conditional probability* that allows us to do just that!

Conditional probability that an event E occurs, given that another event F has already occurred, denoted by $P(E \mid F)$, is defined as

$$P(E|F) = \frac{P(E \cap F)}{P(F)} \quad \text{where} \quad P(F) \neq 0 \tag{1.2}$$

1.2 Conditional Probability and Independence

Conditional probability satisfies the axioms of probability and is a probability measure in the sense of those axioms. Therefore, we can apply any results obtained for a normal probability to a conditional probability. A very useful expression, frequently used in conjunction with the conditional probability, is the so-called *Law of Total Probability*. It says that if $\{A_i \in \Omega, i = 1, 2, \ldots, n\}$ are events such that

(i) $A_i \cap A_j = \varnothing$ *if* $i \neq j$

(ii) $P(A_i) > 0$

(iii) $\bigcup_{i=1}^{n} A_i = \Omega$

1.2 Conditional Probability and Independence

Example 2:

Consider a switching node with three outgoing links A, B and C. Messages arriving at the node can be transmitted over one of them with equal probability. The three outgoing links are operating at different speeds and hence message transmission times are 1, 2 and 3 ms, respectively for A, B and C. Owing to the difference in trucking routes, the probability of transmission errors are 0.2, 0.3 and 0.1, respectively for A, B and C. Calculate the probability of a message being transmitted correctly in 2 ms.

1.2 Conditional Probability and Independence

Solution:

Denote the event that a message is transmitted correctly by F, then we are given

$$P(F \mid A\ Link) = 1 - 0.2 = 0.8$$
$$P(F \mid B\ Link) = 1 - 0.3 = 0.7$$
$$P(F \mid C\ Link) = 1 - 0.1 = 0.9$$

The probability that a message being transmitted correctly in 2 ms is simply the event $(F \cap B)$, hence we have

$$P(F \cap B) = P(F|B) \times P(B)$$
$$= 0.7 \times \frac{1}{3} = \frac{7}{30}$$

1.3 Random Variables and Distributions

In many situations, we are interested in some numerical value that is associated with the outcomes of an experiment rather than the actual outcomes themselves. For example, in an experiment of throwing two die, we may be interested in the sum of the numbers (X) shown on the dice, say $X = 5$. Thus we are interested in a function which maps the outcomes onto some points or an interval on the real line. In this example, the outcomes are {2,3}, {3,2}, {1,4} and {4,1}, and the point on the real line is 5.

This mapping (or function) that assigns a real value to each outcome in the sample space is called a *random variable*. If X is a random variable and x is a real number, we usually write $\{X \leq x\}$ to denote the event $\{\omega \in \Omega$ and $X(\omega) \leq x\}$. There are basically two types of random variables; namely the *discrete random variables* and *continuous random variables*. If the mapping function assigns a real number, which is a point in a countable set of points on the real line, to an outcome then we have a discrete random variable. On the other hand, a continuous random variable takes on a real number which falls in an interval on the real line. In other words, a discrete random variable can assume at most a finite or a countable infinite number of possible values and a continuous random variable can assume any value in an interval or intervals of real numbers.

1.3 Random Variables and Distributions

A concept closely related to a random variable is its *cumulative probability distribution function*, or just *distribution function* (*PDF*). It is defined as

$$\begin{aligned} F_X(x) &\equiv P[X \leq x] \\ &= P[\omega: X(\omega) \leq x] \end{aligned} \tag{1.3}$$

For simplicity, we usually drop the subscript X when the random variable of the function referred to is clear in the context. Students should note that a distribution function completely describes a random variable, as all parameters of interest can be derived from it. It can be shown from the basic axioms of probability that a distribution function possesses the following properties:

1.3 Random Variables and Distributions

(i) F is a non-negative and non-decreasing function, i.e. if $x_1 \le x_2$ then $F(x_1) \le F(x_2)$

(ii) $F(+\infty) = 1$ & $F(-\infty) = 0$

(iii) $F(b) - F(a) = P[a < X \le b]$

For a discrete random variable, its probability distribution function is a dis-int step function, as shown in Figure 1.1. The probability that the random riable takes on a particular value, say x and x = 0, 1, 2, 3 . . . , is given by

$$\begin{aligned} p(x) \equiv P[X = x] &= P[X < x+1] - P[X < x] \\ &= \{1 - P[X \ge x+1]\} - \{1 - P[X \ge x]\} \\ &= P[X \ge x] - P[X \ge x+1] \end{aligned} \qquad (1.4)$$

The above function $p(x)$ is known as the *probability mass function* (*pmf*) of a discrete random variable X and it follows the axiom of probability that

1.3 Random Variables and Distributions

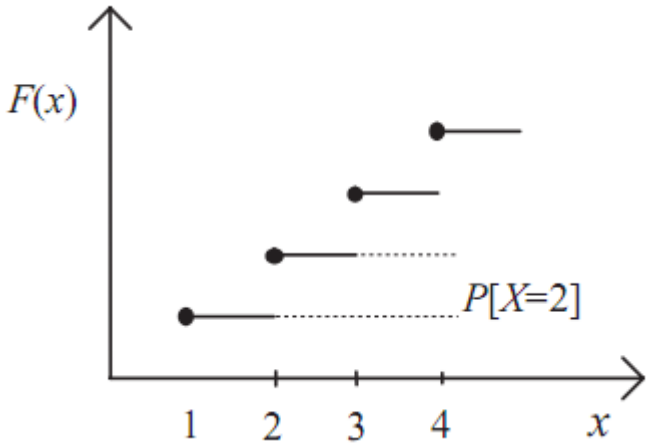


Figure 1.1 Distribution function of a discrete random variable X

1.3 Random Variables and Distributions

We list in this section some important random variables which we will encounter frequently in our subsequent studies of queueing models.

(i) Bernoulli random variable

A *Bernoulli trial* is a random experiment with only two outcomes, 'success' and 'failure', with respective probabilities, p and q. A *Bernoulli random variable* X describes a Bernoulli trial and assumes only two values: 1 (for success) with probability p and 0 (for failure) with probability q:

$$P[X=1]=p \quad \& \quad P[X=0]=q=1-p \tag{1.5}$$

(ii) Binomial random variable

If a Bernoulli trial is repeated k times then the random variable X that counts the number of successes in the k trials is called a *binomial random variable* with parameters k and p. The probability mass function of a binomial random variable is given by

$$B(k;n,p)=\binom{n}{k}p^k q^{n-k} \quad k=0,1,2,\ldots,n \quad \& \quad q=1-p \tag{1.6}$$

1.3 Random Variables and Distributions

(iii) Poisson random variable

A random variable X is said to be *Poisson random variable* with parameter λ if it has the following mass function:

$$P[X = k] = \frac{\lambda^k}{k!} e^{-\lambda} \quad k = 0, 1, 2, \ldots \tag{1.7}$$

1.4 Queueing Systems

In today's information age society, where activities are highly interdependent and intertwined, sharing of resources and hence waiting in queues is a common phenomenon that occurs in every facet of our lives. In the context of data communication, expensive transmission resources in public data networks, such as the Internet, are shared by various network users. Data packets are queued in the buffers of switching nodes while waiting for transmission. In a computer system, computer jobs are queued for CPU or I/O devices in various stages of their processing.

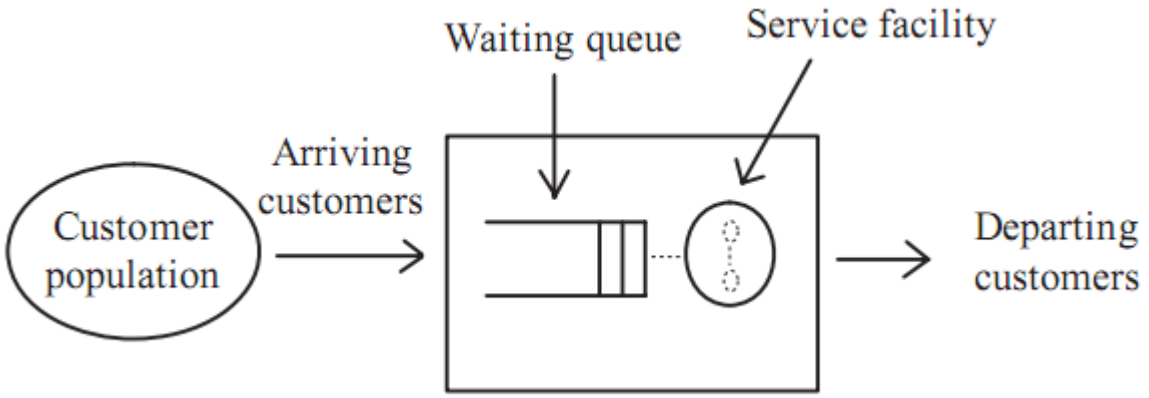


Figure 1.2 Schematic diagram of a queneing system

1.4 Queueing Systems

1. Nomenclature of a queueing system

In the parlance of queueing theory, a *queueing system* is a place where customers arrive according to an 'arrival process' to obtain service from a service facility. The service facility may contain more than one server (or more generally resources) and it is assumed that a server can serve one customer at a time. If an arriving customer finds all servers occupied, he joins a waiting queue. This customer will receive his service later, either when he reaches the head of the waiting queue or according to some service discipline. He leaves the system upon completion of his service.

In the preceding description, we used the generic terms 'customers' and 'servers', which are in line with the terms used in queueing literature. They take various forms in the different domains of applications. In the case of a data switching network, 'customers' are data packets (or data frames or data cells) that arrive at a switching node and 'servers' are the transmission channels. In a CPU job scheduling problem, 'customers' are computer processes (jobs or transactions) and 'servers' are the various computer resources, such as CPU, I/O devices.

1. Nomenclature of a queueing system

So given such a dynamic picture of a queueing system, how do we describe it analytically? How do we formulate a mathematical model that reflects these dynamics? What are the parameters that characterize a queueing system completely? Before we proceed let us examine the structure of a queueing system. Basically, a queueing system consists of three major components:

- The input process
- The system structure
- The output process.

Characteristics of the Input Process

When we talk about the input process, we are in fact concerning ourselves with the following three aspects of the arrival process:

(1) The size of arriving population

The size of the arriving customer population may be infinite in the sense that the number of potential customers from external sources is very large compared to those in the system, so that the arrival rate is not affected by the size.

The size of the arriving population has an impact on the queueing results. An infinite population tends to render the queueing analysis more tractable and often able to provide simple closed-form solutions, hence this model will be assumed for our subsequent queueing systems unless otherwise stated. On the other hand, the analysis of a queueing system with finite customer population size is more involved because the arrival process is affected by the number of customers already in the system.

Examples of infinite customers populations are telephone users requesting telephone lines, and air travellers calling an air ticket reservation system. In fact, these are actually finite populations but they are very large, and for mathematical convenience we treat them as infinite. Examples of the finite calling populations would be a group of stations in a local area network presenting data frame to the broadcast channel, or a group of video display units requesting response from a CPU.

(2) Arriving patterns

Customers may arrive at a queueing system either in some regular pattern or in a totally random fashion. When customers arrive regularly at a fixed interval, the arriving pattern can be easily described by a single number – the rate of arrival. However, if customers arrive according to some random mode, then we need to fit a statistical distribution to the arriving pattern in order to render the queueing analysis mathematically feasible.

The parameter that we commonly use to describe the arrival process is the inter-arrival time between two customers. We generally fit a probability distribution to it so that we can call upon the vast knowledge of probability theory. The most commonly assumed arriving pattern is the Poisson process whose inter-arrival times are exponentially distributed. The popularity of the Poisson process lies in the fact that it describes very well a completely random arrival pattern, and also leads to very simple and elegant queueing results.

We list below some probability distributions that are commonly used to describe the inter-arrival time of an arrival process. These distributions are generally denoted by a single letter as shown:

M: Markovian (or Memoryless), imply Poisson process
D: Deterministic, constant interarrival times
E_k: Erlang distribution of order K of inter-arrival times
G: General probability distribution of inter-arrival times
GI: General and independent (inter-arrival time) distribution.

(3) Behaviour of arriving customers

Customers arriving at a queueing system may behave differently when the system is full due to a finite waiting queue or when all servers are busy. If an arriving customer finds the system is full and leaves forever without entering the system, that queueing system is referred to as a blocking system. The analysis of blocking systems, especially in the case of queueing networks, is more involved and at times it is impossible to obtain closed-form results. We will assume that this is the behaviour exhibited by all arriving customers in our subsequent queueing models. In real life, customers tend to come back after a short while.

Characteristics of the System Structure

(i) Physical number and layout of servers

The service facility shown in Figure 1.2 may comprise of one or more servers. In the context of this book, we are interested only in the parallel and identical servers; that is a customer at the head of the waiting queue can go to any server that is free, and leave the system after receiving his/her service from that server, as shown in Figure 1.3 (a). We will not concern ourselves with the serial-servers case where a customer receives services from all or some of them in stages before leaving the system, as shown in Figure 1.3 (b).

(ii) The system capacity

The system capacity refers to the maximum number of customers that a queueing system can accommodate, inclusive of those customers at the service facility.

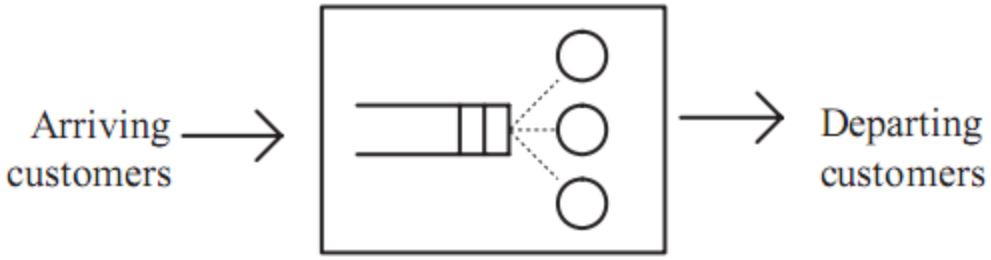


Figure 1.3 (a) Parallel servers

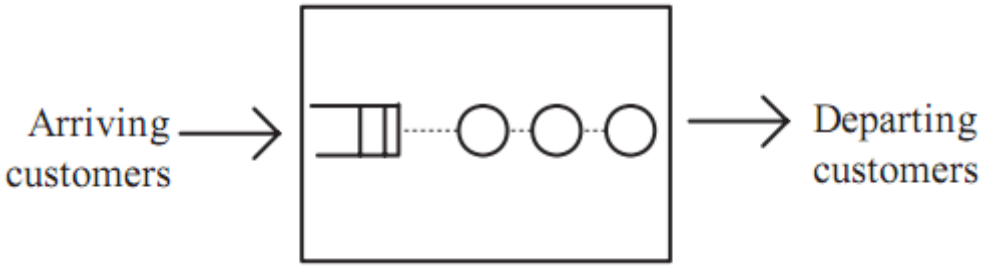


Figure 1.3 (b) Serial servers

In a multi-server queueing system, as shown in Figure 1.3 (a), the system capacity is the sum of the maximum size of the waiting queue and the number of servers. If the waiting queue can accommodate an infi nite number of customers, then there is no blocking, arriving customers simply joining the waiting queue. If the waiting queue is fi nite, then customers may be turned away. It is much easier to analyse queueing systems with infi nite system capacity as they often lead to power series that can be easily put into closed form expressions.

Characteristics of the Output Process

(i) Queueing discipline or serving discipline

Queueing discipline, sometimes known as serving discipline, refers to the way in which customers in the waiting queue are selected for service. In general, we have:

- First-come-first-served (FCFS)
- Last-come-first-served (LCFS)
- Priority
- Processor sharing
- Random.

The FCFS queueing discipline does not assign priorities and serves customers in the order of their arrivals. Apparently this is the most frequently encountered discipline at an ordered queue and therefore it will be the default queueing discipline for all the subsequent systems discussed, unless otherwise stated.

The LCFS discipline is just the reverse of FCFS. Customers who come last will be served fi rst. This type of queueing discipline is commonly found in stack operations where items (customers in our terminology) are stacked and operations occur only at the top of the stack.

In priority queueing discipline, customers are divided into several priority classes according to their assigned priorities. Those having a higher priority than others are served fi rst before others receiving their service. There are two sub-classifi cations: preemptive and non-preemptive.

In processor sharing, capacity is equally divided among all customers in the queue, that is when there are k customers, the server devotes 1/k of his capacity to each. Equally, each customer obtains service at 1/k of rate and leaves the system upon completion of his service.

(ii) Service-time distribution

Similar to arrival patterns, if all customers require the same amount of service time then the service pattern can be easily described by a single number. But generally, different customers require different amounts of service times, hence we again use a probability distribution to describe the length of service times the server renders to those customers. The most commonly assumed service time distribution is the negative exponential distribution.

Again, we commonly use a single letter to indicate the type of service distributions:

M: Markovian (or Memoryless) , imply exponential distributed service times
D: Deterministic ; constant service times
E_k: Erlang of order K service time distribution
G: General service times distribution.

2. Random variables and their relationships

From the preceding description of a queueing system, we see that customers arrive, are served and leave the system, hence presenting a fl uid situation with constant motions. There are many processes present and interacting with each other. Most of the quantities associated with these processes evolve in time and are of a probabilistic nature. In other words, they are the so-called random variables and their values can only be expressed through probability. We summarize the primary random variables of a queueing system in Table 1.1 and list some of the relationships among them.

Table 1.1 Random variables of a queueing system

Notation	Description
$N(t)$	The number of customers in the system at time t
$N_q(t)$	The number of customers in the waiting queue at time t
$N_s(t)$	The number of customers in the service facility at time t
N	The average number of customers in the system
N_q	The average number of customers in the waiting queue
N_s	The average number of customers in the service facility
T_k	The time spent in the system by kth customer
W_k	The time spent in the waiting queue by kth customer
x_k	The service time of kth customer
T	The average time spent in the system by a customer
W	The average time spent in the waiting queue by a customer
$\bar{x}$	The average service time
$P_k(t)$	The probability of having k customers in the system at time t
P_k	The stationary probability of having k customers in the system

Looking at the structure of a queueing system, we can easily arrive at the following expressions:

$$N(t) = N_q(t) + N_s(t) \quad \text{and} \quad N = N_q + N_s \tag{1.7}$$

$$T_k = W_k + x_k \quad \text{and} \quad T = W + \overline{x} \tag{1.8}$$

3. Kendall notation

From the above section, we see that there are many stochastic processes and a multiplicity of parameters (random variables) involved in a queueing system, so given such a complex situation how do we categorize them and describe them succinctly in a mathematical short form? David G Kendall, a British statistician, devised a shorthand notation, shown below, to describe a queueing system containing a single waiting queue. This notation is known as Kendall notation:

$$A / B / X / Y / Z$$

where
- A : Customer arriving pattern (Inter-arrival-time distribution)
- B : Service pattern (Service-time distribution)
- X : Number of parallel servers
- Y : System capacity
- Z : Queueing discipline

Example:

Figure 1.4 shows a computer setup where a pool of m remote-job-entry terminals is connected to a central computer. Each operator at the terminals spends an average of S seconds thinking and submitting a job (or request) that requires P seconds at the CPU. These submitted jobs are queued and later processed by the single CPU in an unspecifi ed queueing discipline.
We would like to estimate the maximum throughput of the system, so propose a queueing model that allows us to do so.

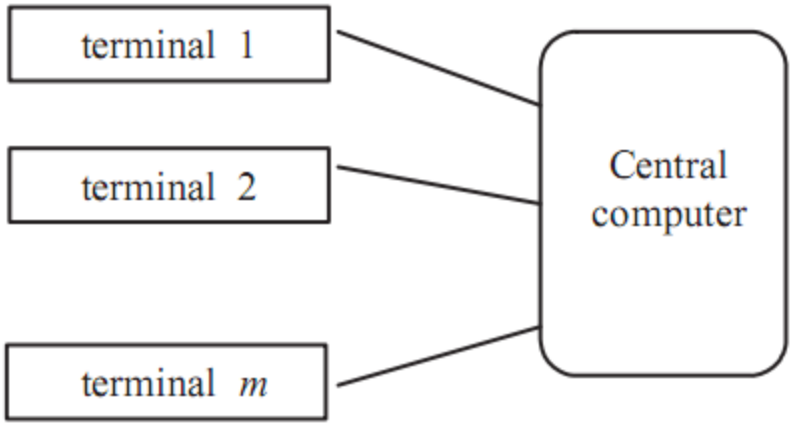


Figure 1.4 A job-processing system

Solution

Firstly, we assume that the thinking times (or job submission times) and CPU processing times are exponentially distributed, hence the arrival process is Poisson. Secondly, we further assume that those operators at the terminals are either waiting for their responses from the CPU or actively entering their requests, so if there are k requests waiting to be processed there are $(m - k)$ terminals active in the arrival process. Thus we can represent the jobs entered as a state-dependent Poisson process with the rate:

$$\lambda(k) = \begin{cases} (m-k)\lambda & k < m \\ 0 & k \geq m \end{cases} \qquad (1.9)$$

1.5 Little's theorem

Before we examine the stochastic behaviour of a queueing system, let us first establish a very simple and yet powerful result that governs its steady-state performance measures – *Little's theorem, law* or *result* are the various names. This result existed as an empirical rule for many years and was first proved in a formal way by J D C Little in 1961 (Little 1961).

This theorem relates the average number of customers (N) in a steady-state queueing system to the product of the average arrival rate (λ) of customers entering the system and the average time (T) a customer spent in that system, as follows:

$$N = \lambda T \tag{1.10}$$

This result was derived under very general conditions. The beauty of it lies in the fact that it does not assume any specific distribution for the arrival as well as the service process, nor it assumes any queueing discipline or depends upon the number of parallel servers in the system. With proper interpretation of N, λ and T, it can be applied to all types of queueing systems, including priority queueing and multi-server systems.

Here, we offer a simple proof of the theorem for the case when customers are served in the order of their arrivals. In fact, the theorem holds for any queueing disciplines as long as the servers are kept busy when the system is not empty.

Let us count the number of customers entering and leaving the system as functions of time in the interval $(0, t)$ and define the two functions:

$A(t)$: Number of arrivals in the time interval $(0, t)$
$D(t)$: Number of departures in the time interval $(0, t)$

Assuming we begin with an empty system at time 0, then $N(t) = A(t) - D(t)$ represents the total number of customers in the system at time t. A general sample pattern of these two functions is depicted in Figure 1.5, where t_k is the instant of arrival and T_k the corresponding time spent in the system by the k^{th} customer.

$$\int_0^t N(\tau)d\tau = \int_0^t [A(\tau) - D(\tau)]d\tau$$

$$= \sum_{k=1}^{D(t)} T_k \times 1 + \sum_{k=D(t)+1}^{A(t)} (t - t_k) \times 1$$

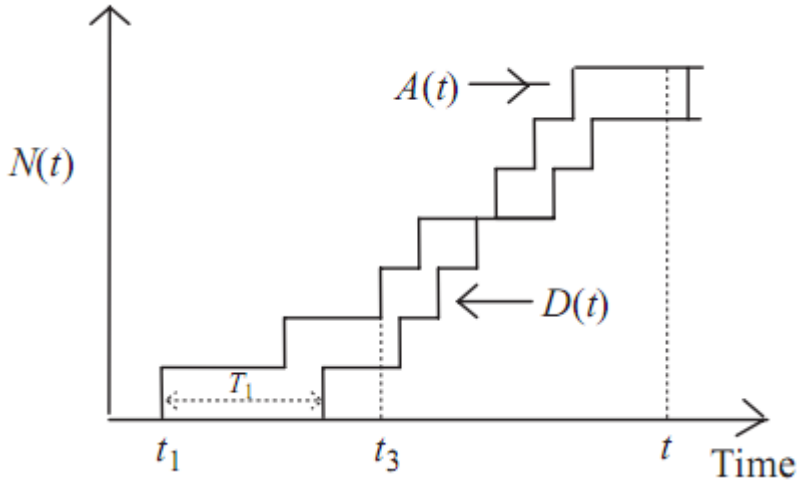


Figure 1.5 A sample pattern of arrival

hence $$\frac{1}{t}\int_0^t N(\tau)d\tau = \left[\sum_{k=1}^{D(t)} T_k + \sum_{k=D(t)+1}^{A(t)} (t - t_k) \right] \times \frac{1}{A(t)} \times \frac{A(t)}{t} \qquad (1.11)$$

We recognize that the left-hand side of the equation is simply the *time-average* of the number of customers (N_t) in the system during the interval $(0, t)$, whereas the last term on the right $A(t)/t$ is the time-average of the customer arrival rate (λ_t) in the interval $(0, t)$. The remaining bulk of the expression on the right:

$$\left[\sum_{i=1}^{D(t)} T_i + \sum_{i=D(t)+1}^{A(t)} (t - t_i)\right] \Big/ A(t)$$

is the time-average of the time (T_t) a customer spends in the system in $(0, t)$. Therefore, we have

$$N_t = \lambda_t \times T_t \qquad (1.12)$$

Students should recall from probability theory that the number of customers in the system can be computed as

$$N = \sum_{k=0}^{\infty} kP_k$$

The quantity N computed in this manner is the ensemble (or stochastic) average or the so-called expected value. We mentioned that ensemble averages are equal to time averages for an ergodic system, and in general most of the practical queueing systems encountered are ergodic, thus we have

$$N = \lambda \times T \tag{1. 13}$$

where λ and T are ensemble quantities.

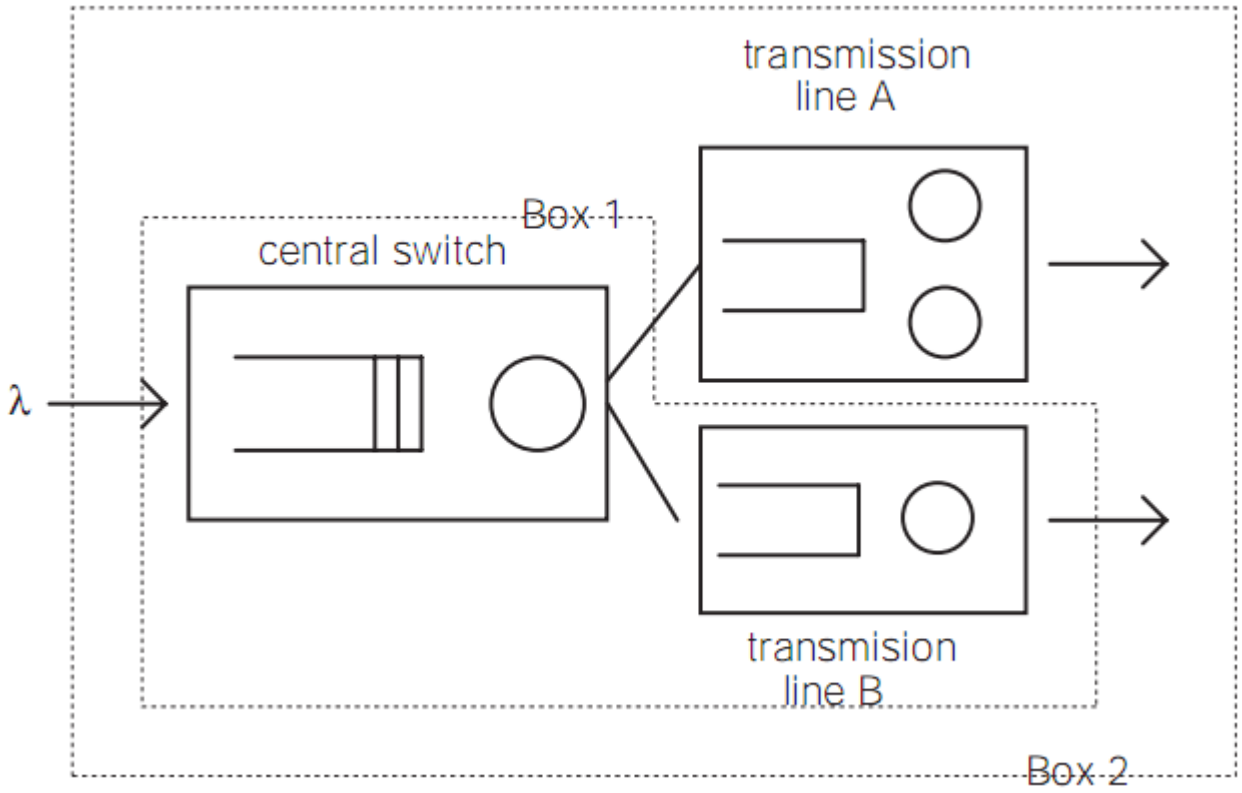


Figure 1.6 A queueing model of a switch

This result holds for both the time average quantities as well as stochastic (or ensemble) average quantities. We shall clarify further the concept of ergodicity later and we shall assume that all queueing systems we are dealing with are ergodic in subsequent chapters.

1.5.1. General Applications of Little's Theorem

Little's formula was derived in the preceding section for a queuing system consisting of only one server. It is equally valid at different levels of abstraction, for example the waiting queue, or a larger system such as a queueing network. To illustrate its applications, consider the hypothetical queueing network model of a packet switching node with two outgoing links, as depicted in Figure 1.6.

If we apply Little's result to the waiting queue and the service facility of the central switch respectively, we have

$$N_q = \lambda W \qquad (1.\ 14)$$

$$N_s = \lambda \overline{x} = \rho \qquad (1.\ 15)$$

where N_q is the number of packets in the waiting queue and N_s is the average number of packets at the service facility. We introduce here a new quantity ρ, a utilization factor, which is always less than one for a stable queueing system.

Box 1 illustrates the application to a sub-system which comprises the central switch and transmission line B. Here the time spent in the system (T_{sub}) is the total time required to traverse the switch and the transmission line B. The number of packets in the system (N_{sub}) is the sum of packets at both the switch and the transmission line B:

$$N_{sub} = N_{sw} + N_B = \lambda T_{sw} + \lambda_B T_B$$

$$T_{sub} = T_{sw} + T_B \qquad (1.16)$$

where T_{sw} is the time spent at the central switch
T_B is the time spent by a packet to be transmitted over Line B
N_{sw} is the number of packets at the switch
N_B is the number of packets at Line B
λ_B is the arrival rate of packets to Line B.

We can also apply Little's result to the system as a whole (Box 2). Then T and N are the corresponding quantities of that system:

$$N = \lambda T \qquad (1.17)$$

1.6.Resource utilization and traffic intensity

The concept of resource utilization, sometimes referred to as utilization factor (or just utilization), is basically a measure of how busy a resource is. In the context of queueing theory, it represents the fraction of time a server is engaged in providing service, defined as

$$\text{Utilization } (\rho) = \frac{\text{Time a server is occupied}}{\text{Time available}}$$

In the case of a queueing system with $m (m \geq 1)$ servers, if there are N customers in the system within a time interval $(t, t + T)$, then each server on average will serve $(\lambda T)/m$ customers. If the average service time is $\bar{x} = 1/\mu$ unit of time, then we have

$$\rho = \frac{(\lambda T/m) \times (1/\mu)}{T} = \frac{\lambda}{m\mu} \qquad (1.16)$$

It is clear from the above definition that ρ is dimensionless and should be less than unity in order for a server to cope with the service demand, or in other words for the system to be stable.

A measure that is commonly used in traffic engineering and closely related to the concept of resource utilization is *traffic intensity*. It is a measure of the total arrival traffic presented to the system as a whole. It is also known as *offered load* and is by definition just the product of average arrival rate and the average service time:

$$\text{Traffic intensity} \quad (\alpha) = \lambda \bar{x} = \lambda/\mu \tag{1.17}$$

In fact, traffic intensity is again a dimensionless quantity but it is usually expressed in erlangs, named after A K Erlang.

To better understand the physical meaning of this unit, take a look at the traffic presented to a single resource. One erlang of traffic is equivalent to a single user who uses that resource 100% of the time or alternatively, 10 users who each occupy the resource 10% of the time. A traffic intensity greater than one indicates that customers arrive faster than they are served and is a good indication of the minimum number of servers required to achieve a stable system. For example, a traffic intensity of 2.5 erlangs indicates that at least 3 servers are required.

1.7. Flow conservation law

This is analogous to Kirchhoff's current law which states that the algebraic sum of all the electrical currents entering a node must equal the algebraic sum of all the currents leaving a node.

For a stable queueing system, the rate of customers entering the system should equal the rate of customers leaving the system, if we observe it for a suffi ciently long period of time, i.e. $\lambda_{out} = \lambda_{in}$, as shown in Figure 1.7. This is initiatively clear because if $\lambda_{out} < \lambda_{in}$ then there will be a steady build-up of customers and the system will eventually become unstable. On the other hand, if $\lambda_{out} > \lambda_{in}$, then customers are created within the system.

This notion of fl ow conservation is useful when we wish to calculate through-put. It can be applied to individual queued in a collection of queueing systems.

Example

(a) A communication channel operating at 9600 bps receives two types of packet streams from a gateway. Type A packets have a fixed length format of 48 bits whereas Type B packets have an exponentially distributed length with a mean of 480 bits. If on average there are 20% Type A packets and 80% Type B packets, calculate the utilization of this channel assuming the combined arrival rate is 15 packets per second.

(b) A PBX was installed to handle the voice traffic generated by 300 employees in a factory. If each employee, on average, makes 1.5 calls per hour with

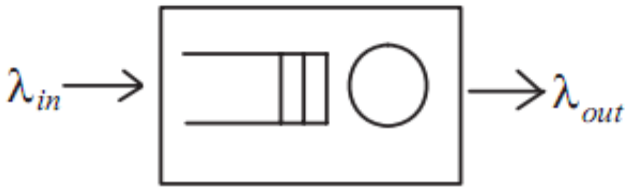


Figure 1.7 Flow Conservation Law

an average call duration of 2.5 minutes, what is the offered load presented to this PBX?

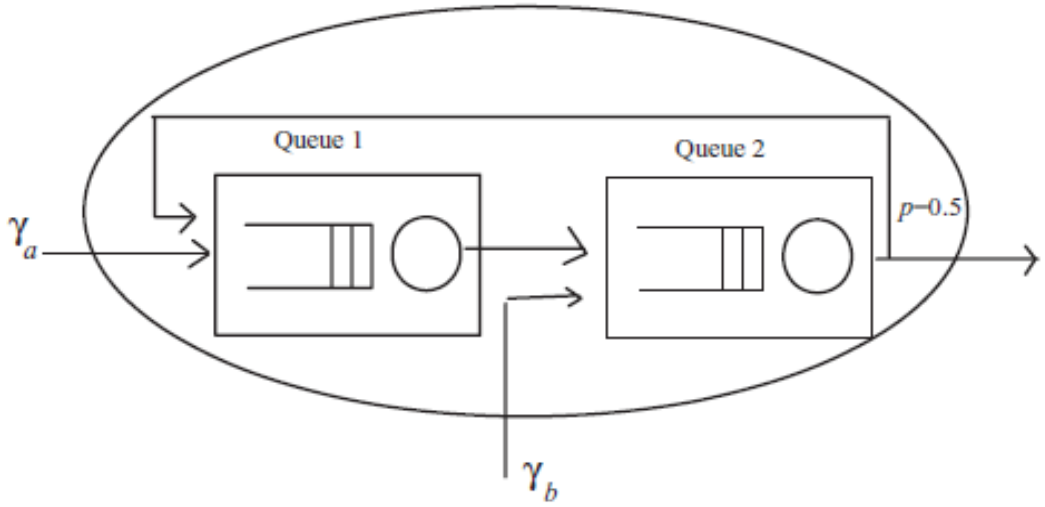
Solution

(a) The average transmission time
$$= (0.2 \times 48 + 0.8 \times 480)/9600$$
$$= 0.041 \text{ s}$$
$$\rho = 15 \times 0.041 = 61.5\%$$

(b) *Offered load* = *Arrival rate* × *Service time*
$$= 300 \text{ (}\textit{users}\text{)} \times 1.5 \text{ (}\textit{calls per user per hour}\text{)}$$
$$\times 2.5 \text{ (minutes } \textit{percall}\text{)} \div 60 \text{ (minutes } \textit{perhour}\text{)}$$
$$= 18.75 \textit{ erlangs}$$

Example

Consider the queueing system shown below where we have customers arriving to Queue 1 and Queue 2 with rates γ_a and γ_b, respectively. If the branching probability p at Queue 2 is 0.5, calculate the effective arrival rates to both queues.



Solution

Denote the effective arrival rates to Queue 1 and Queue 2 as λ_1 and λ_2, respectively, then we have the following expressions under the principal of flow conservation:

$$\lambda_1 = \gamma_a + 0.5\lambda_2$$

$$\lambda_2 = \lambda_1 + \gamma_b$$

Hence, we have

$$\lambda_1 = 2\gamma_a + \gamma_b$$

$$\lambda_2 = 2(\gamma_a + \gamma_b)$$

1.8. Poisson process

Poisson process is central to physical process modelling and plays a pivotal role in classical queuing theory. In most elementary queueing systems, the inter-arrival times and service times are assumed to be exponentially distributed or, equivalently, that the arrival and service processes are Poisson, as we shall see below. The reason for its ubiquitous use lies in the fact that it possesses a number of marvellous probabilistic properties that give rise to many elegant queueing results. Secondly, it also closely resembles the behaviour of numerous physical phenomenon and is considered to be a good model for an arriving process that involves a large number of similar and independent users.

Owing to the important role of Poisson process in our subsequent modelling of arrival processes to a queueing system, we will take a closer look at it and examine here some of its marvellous properties.

Put simply, a Poisson process is a counting process for the number of randomly occurring point events observed in a given time interval $(0, t)$. It can also be deemed as the limiting case of placing at random k points in the time interval of $(0, t)$. If the random variable $X(t)$ that counts the number of point events in that time interval is distributed according to the well-known Poisson distribution given below, then that process is a Poisson process:

$$P[X(t)=k]=\frac{(\lambda t)^k}{k!}e^{-\lambda t} \qquad (1.\ 18)$$

Here, λ is the rate of occurrence of these point events and λt is the mean of a Poisson random variable and physically it represents the average number of occurrences of the event in a time-interval t. Poisson distribution is named after the French mathematician, Simeon Denis Poisson.

1.8.1. The poisson process – a limiting case

The Poisson process can be considered as a limiting case of the Binomial distribution of a Bernoulli trial. Assuming that the time interval $(0, t)$ is divided into time slots and each time slot contains only one point, if we place points at random in that interval and consider a point in a time slot as a 'success', then the number of k 'successes' in n time slots is given by the Binomial distribution:

$$P[k \text{ successes in } n \text{ time-slots}] = \binom{n}{k} p^k (1-p)^{n-k}$$

Now let us increase the number of time slots (n) and at the same time decrease the probability (p) of 'success' in such a way that the average number of 'successes' in a time interval t remains constant at $np = \lambda t$, then we have the Poisson distribution:

$$P[k \text{ arrivals in } (0, t)] = \frac{(\lambda t)^k}{k!} e^{-\lambda t} \qquad (1.19)$$

1.8.2. Properties of the poisson process

a. Superposition property

The superposition property says that if k independent Poisson processes A1, A2, . . . Ak are combined into a single process A = A1 + A2 + . . . + Ak, then A is still Poisson with rate l equal to the sum of the individual rates li of Ai, as shown in Figure 1.8.

Recall that the z-transform of a Poisson distribution with parameter λt is $e^{-\lambda t(1-z)}$.

Since $$A = A_1 + A_2 + \cdots + A_k$$

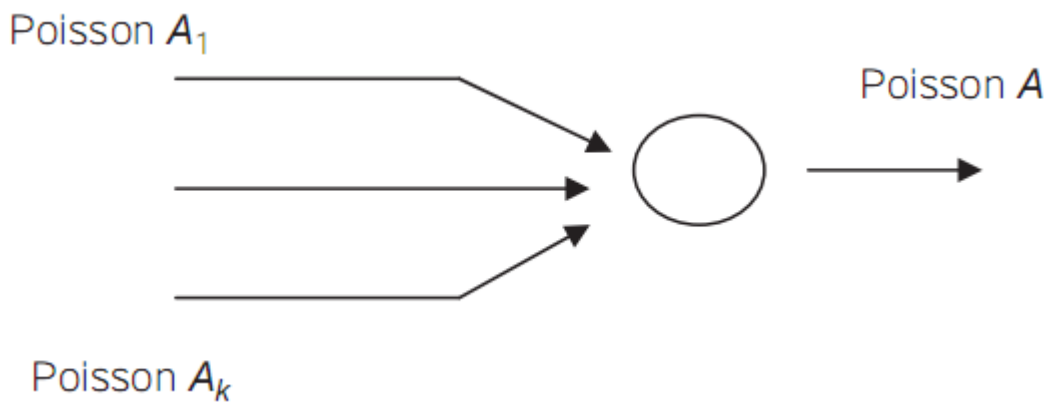


Figure 1.8 Superposition property

1.8.2. Properties of the poisson process

b. Decomposition property

The decomposition property is just the reverse of the previous property, as shown in Figure 2.12, where a Poisson process A is split into k processes using probability p_i $(i = 1, \ldots, k)$.

Let us derive the probability mass function of a typical process A_i. On condition that there are N arrivals during the time interval $(0, t)$ from process A, the probability of having k arrivals at process A_i is given by

$$P[A_i(t) = k] = \frac{(p_i \lambda t)^k}{k!} e^{-p_i \lambda t} \qquad (1.\ 20)$$

That is, a Poisson process with rate $p_i\lambda$.

1.8.2. Properties of the poisson process

c. Exponentially distributed inter-arrival times

The exponential distribution and the Poisson process are closely related and in fact they mirror each other in the following sense. If the inter-arrival times in a point process are exponentially distributed, then the number of arrival points in a time interval is given by the Poisson distribution and the process is a Poisson arrival process. Conversely, if the number of arrival points in any interval is a Poisson random variable, the inter-arrival times are exponential distributed and the arrival process is Poisson.

Let τ be the inter-arrival time, then

$$P[\tau \leq t] = 1 - P[\tau > t].$$

But $P[\tau > t]$ is just the probability that no arrival occurs in $(0, t)$; i.e. $P_0(t)$. Therefore we obtain

$$P[\tau \leq t] = 1 - e^{-\lambda t} \quad \text{(exponential distribution)} \qquad (1.21)$$

1.8.2. Properties of the poisson process

d. Memoryless (markovian) property of inter-arrival times

The memoryless property of a Poisson process means that if we observe the process at a certain point in time, the distribution of the time until next arrival is not affected by the fact that some time interval has passed since the last

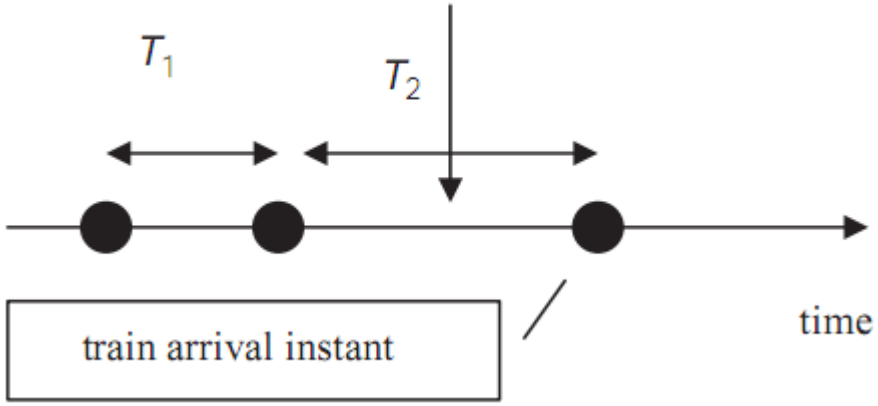


Figure 1.9 Sample train arrival instants

arrival. In other words, the process starts afresh at the time of observation and has no memory of the past. Before we deal with the formal definition, let us look at an example to illustrate this concept.

1.8.2. Properties of the poisson process

Example

Consider the situation where trains arrive at a station according to a Poisson process with a mean inter-arrival time of 10 minutes. If a passenger arrives at the station and is told by someone that the last train arrived 9 minutes ago, so on the average, how long does this passenger need to wait for the next train?

1.8.2. Properties of the poisson process

Solution

Intuitively, we may think that 1 minute is the answer, but the correct answer is 10 minutes. The reason being that Poisson process, and hence the exponential inter-arrival time distribution, is memoryless. What have happened before were sure events but they do not have any influence on future events.

Mathematically, the 'memoryless' property states that the distribution of remaining time until the next arrival, given that t0 units of time have elapsed since the last arrival, is identically equal to the unconditional distribution of inter-arrival times (Figure 1.10). Assume that we start observing the process immediately after an arrival at time 0. From Equation above we know that the probability of no arrivals in (0, t0) is given by

$$P[no\ arrival\ in\ (0, t_0)] = e^{-\lambda t_0}$$

1.8.2. Properties of the poisson process

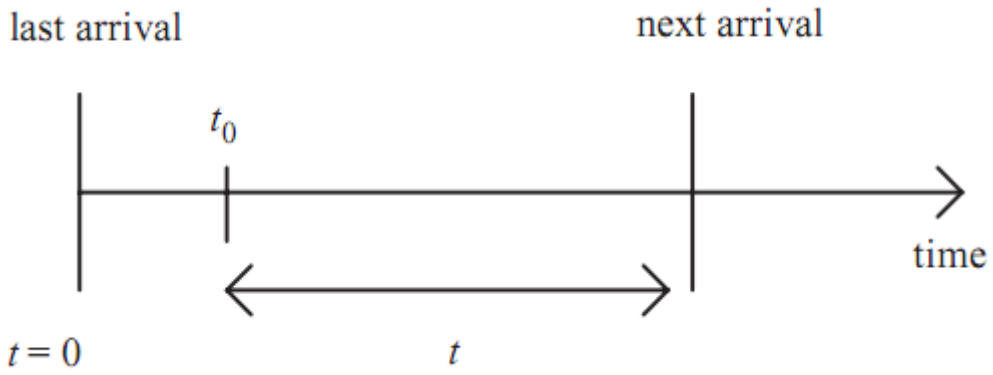


Figure 1.10 Conditional inter-arrival times

Let us now find the conditional probability that the first arrival occurs in $[t_0, t_0 + t]$, given that t_0 has elapsed; that is

$$P[arrival\ in\ (t_0, t_0+t) | no\ arrival\ in (0, t_0)] = \frac{\int_{t_0}^{t_0+t} \lambda e^{-\lambda t}}{e^{-\lambda t_0}} = 1 - e^{-\lambda t}$$

But the probability of an arrival in $(0, t)$ is also $\int_0^t \lambda e^{-\lambda t} dt = 1 - e^{-\lambda t}$

1.8.2. Properties of the poisson process

Example

Let us consider again the problem presented in Example above. When this passenger arrives at the station:

a) What is the probability that he will board a train in the next 5 minutes?
b) What is the probability that he will board a train in 5 to 9 minutes?

1.8.2. Properties of the poisson process

Solution

a) From example, we have **λ** = 1/10 = 0.1 min−1, hence for a time period of 5 minutes we have

$$\lambda t = 5 \times 0.1 = 0.5 \quad \text{and}$$

$$P[0\ train\ in\ 5\,\text{min}] = \frac{e^{-\lambda t}(\lambda t)^k}{k!} = \frac{e^{-0.5}(0.5)^0}{0!} = 0.607$$

He will board a train if at least one train arrives in 5 minutes; hence

$$P[at\ least\ 1\ train\ in\ 5\ \text{min}] = 1 - P[0\ train\ in\ 5\ \text{min}]$$
$$= 0.393$$

1.8.2. Properties of the poisson process

b) He will need to wait from 5 to 9 minutes if no train arrives in the first 5 minutes and board a train if at least one train arrives in the time interval 5 to 9 minutes. From (a) we have

$$P[0 \textit{ train in } 5 \text{ min}] = 0.607$$

and $P[\textit{at least } 1 \textit{ train in next } 4 \text{ min}] = 1 - P[0 \textit{ train in next } 4 \text{ min}]$

$$= 1 - \frac{e^{-0.4}(0.4)^0}{0!} = 0.33$$

Hence, $P[0 \textit{ train in } 5 \text{ min } \& \textit{ at least } 1 \textit{ train in next } 4 \text{ min}]$
$= P[0 \textit{ train in } 5 \text{ min}] \times P[\textit{at least } 1 \textit{ train in next } 4 \text{ min}]$
$= 0.607 \times 0.33 = 0.2$

Chapter 2. QUEUEING THEORY

1. Discrete and Continuous Markov Processes

Before we embark on the theory of Markov processes, let us look at the more general random processes – the so-called stochastic processes. The Markov process is a special class of stochastic processes that exhibits particular kinds of dependencies among the random variables within the same process. It provides the underlying theory of analysing queueing systems. In fact, each queueing system can, in principle, be mapped onto an instance of a Markov process or its variants (e.g. Imbedded Markov process) and mathematically analysed. We shall discuss this in detail later in the chapter.

1.1 Stochastic processes

Simply put, a *stochastic process* is a mathematical model for describing an empirical process that changes with an index, which is usually the time in most of the real-life processes, according to some probabilistic forces. More specifically, a stochastic process is a family of random variables $\{X(t), t \in T\}$ defined on some probability space and indexed by a parameter $t\{t \in T\}$, where t is usually called the time parameter. The probability that $X(t)$ takes on a value, say i and that is $P[X(t) = i]$, is the range of that probability space.

In our daily life we encounter many stochastic processes. For example, the price $P_{st}(t)$ of a particular stock counter listed on the Singapore stock exchange as a function of time is a stochastic process. The fluctuations in $P_{st}(t)$ throughout the trading hours of the day can be deemed as being governed by probabilistic forces and hence a stochastic process. Another example will be the number of customers calling at a bank as a function of time.

Basically, there are three parameters that characterize a stochastic process:

1.1 Stochastic processes

(1) State space

The values assumed by a random variable $X(t)$ are called '*states*' and the collection of all possible values forms the '*state space*' of the process. If $X(t) = i$ then we say the process is in state i. In the stock counter example, the state space is the set of all prices of that particular counter throughout the day.

If the state space of a stochastic process is finite or at most countably infinite, it is called a '*discrete-state*' process, or commonly referred to as a stochastic chain. In this case, the state space is often assumed to be the non-negative integers $\{0, 1, 2, \ldots\}$. The stock counter example mentioned above is a discrete-state stochastic chain since the price fluctuates in steps of few cents or dollars.

1.1 Stochastic processes

On the other hand, if the state space contains a finite or infinite interval of the real numbers, then we have a '*continuous-state*' process. At this juncture, let us look at a few examples about the concept of 'countable infinite' without going into the mathematics of set theory. For example, the set of positive integer numbers $\{n\}$ in the interval $[a, b]$ is finite or countably infinite, whereas the set of real numbers in the same interval $[a, b]$ is infinite.

In the subsequent study of queueing theory, we are going to model the number of customers in a queueing system at a particular time as a Markov chain and the state represents the actual number of customers in the system. Hence we will restrict our discussion to the discrete-space stochastic processes.

1.1 Stochastic processes

(2) Index parameter

As mentioned above, the index is always taken to be the time parameter in the context of applied stochastic processes. Similar to the state space, if a process changes state at discrete or finite countable time instants, we have a '*discrete (time) – parameter*' process. A discrete-parameter process is also called a *stochastic sequence*. In this case, we usually write $\{X_k \mid k \in N = (0, 1, 2, \ldots)\}$ instead of the enclosed time parameter $\{X(t)\}$. Using the stock price example again, if we are only interested in the closing price of that counter then we have a stochastic sequence.

On the other hand, if a process changes state (or in the terminology of Markov theory makes a '*transition*') at any instant on the time axis, then we have a '*continuous (time) – parameter*' process. For example, the number of arrivals of packets to a router during a certain time interval $[a, b]$ is a continuous-time stochastic chain because $t \in [a, b]$ is a continuum.

1.1 Stochastic processes

(3) Statistical dependency

Statistical dependency of a stochastic process refers to the relationships between one random variable and others in the same family. It is the main feature that distinguishes one group of stochastic processes from another.

Any realization of a stochastic process is called a sample path. For example, a sample path of tossing a coin n times is {head, tail, tail, head, . . . , head}.

Markov processes are stochastic processes which exhibit a particular kind of dependency among the random variables. For a Markov process, its future probabilistic development is dependent only on the most current state, and how the process arrives at the current position is irrelevant to the future concern. More will be said about this process later.

In the study of stochastic processes, we are generally interested in the probability that $X(t)$ takes on a value i at some future time t, that is $\{P[X(t) = i]\}$, because precise knowledge cannot be had about the state of the process in future times. We are also interested in the steady state probabilities if the probability converges.

1.1 Stochastic processes

Example

Let us denote the day-end closing price of a particular counter listed on the Singapore stock exchange on day k as X_k. If we observed the following closing prices from day k to day $k + 3$, then the following observed sequence $\{X_k\}$ is a stochastic sequence:

$$X_k = \$2.45 \quad X_{k+1} = \$2.38$$

$$X_{k+2} = \$2.29 \quad X_{k+3} = \$2.78$$

However, if we are interested in the fluctuations of prices during the trading hours and assume that we have observed the following prices at the instants $t_1 < t_2 < t_3 < t_4$, then the chain $\{X(t)\}$ is a continuous-time stochastic chain:

2. Discrete-time markov chains

The *discrete-time Markov chain* is easier to conceptualize and it will pave the way for our later introduction of the continuous Markov processes, which are excellent models for the number of customers in a queueing system.

exhibits a simple but very useful form of dependency among the random variables of the same family, namely the dependency that each random variable in the family has a distribution that depends only on the immediate preceding random variable. This particular type of dependency in a stochastic process was first defined and investigated by the Russian mathematician Andrei A Markov and hence the name Markov process, or Markov chain if the state space is discrete. In the following sections we will be merely dealing with only the discrete-state Markov processes; we will use Markov chain or process interchangeably without fear of confusion.

As an illustration to the idea of a Markov chain vs a stochastic process, let us look at a coin tossing experiment. Firstly, let us define two random variables; namely $X_k = 1$ (or 0) when the outcome of the kth trial is a 'head' (or tail), and Y_k = accumulated number of 'heads' so far. Assuming the system starts in state Zero ($Y_0 = 0$) and has the following sequence of the outcomes

2.1. Defi nition of Discrete-time Markov Chains

Mathematically, a stochastic sequence $\{X_k, k \in T\}$ is said to be a discrete-time Markov chain if the following conditional probability holds for all i, j and k:

$$P[X_{k+1} = j \mid X_0 = i_0, X_1 = i_1, \ldots, X_{k-1} = i_{k-1}, X_k = i] = P[X_{k+1} = j \mid X_k = i] \quad (2.1)$$

The above expression simply says that the $(k + 1)$th probability distribution conditional on all preceding ones equals the $(k + 1)$th probability distribution conditional on the kth; $k = 0, 1, 2, \ldots$. In other words, the future probabilistic development of the chain depends only on its current state (kth instant) and not on how the chain has arrived at the current state. The past history has been completely summarized in the specification of the current state and the system has no memory of the past – a '*memoryless*' chain. This '*memoryless*' characteristic is commonly known as the *Markovian* or *Markov property*.

2.1. Defi nition of Discrete-time Markov Chains

The conditional probability at the right-hand side of Equation (2.1) is the probability of the chain going from state i at time step k to state j at time step $k + 1$ – the so-called *(one-step) transitional probability*. In general $P[Xk+1 = j \,|Xk = i]$ is a function of time and in Equation (2.1) depends on the time step k. If the transitional probability does not vary with time, that is, it is invariant with respect to time epoch, then the chain is known as a *time-homogeneous Markovchain*. Using a short-hand notation, we write the conditional probability as

$$p_{ij} = P[X_{k+1} = j \mid X_k = i] \tag{2.2}$$

dropping the time index. Throughout this book, we will assume that all Markov processes that we deal with are time homogeneous.

For notational convention, we usually denote the state space of a Markov chain as $\{0, 1, 2, \ldots\}$. When $X_k = j$, the chain is said to be in state j at time k and we define the probability of finding the chain in this state using the new notation:

2.1. Defi nition of Discrete-time Markov Chains

$$\pi_j^{(k)} \equiv P[X_k = j]$$

When a Markov chain moves from one state to another, we say the system makes a '*transition*'. The graphical representation of these dynamic changes of state, as shown in Figure 2.1, is known as the '*state transition diagram*' or '*transition diagram*' for short. In the diagram, the nodes represent states and the directed arcs between nodes represent the one-step transition probabilities. Those self-loops indicate the probabilities of remaining in the same state at the next time instant. In the case of a discrete-time Markov chain, the transitions between states can take place only at some integer time instants 0, 1, 2, . . . *k*, whereas transitions in the continuous case may take place at any instant of time.

As the system must transit to another state or remain in its present state at the next time step, we have

2.1. Defi nition of Discrete-time Markov Chains

$$\sum_j p_{ij} = 1 \quad \& \quad 1 - p_{ii} = \sum_{j \neq i} p_{ij} \tag{2.3}$$

The conditional probability shown in Equation (2.1) expresses only the dynamism (or movement) of the chain. To characterize a Markov chain completely, it is necessary to specify the starting point of the chain, or in other words, the initial probability distribution $P[X0 = i]$ of the chain. Starting with the initial state, it is in principle now possible to calculate the probabilities of fi nding the chain in a particular state at a future time using the total probability theorem:

$$P[X_{k+1} = j] = \sum_{i=0}^{\infty} P[X_{k+1} = j | X_k = i] P[X_k = i] = \sum_{i=0}^{\infty} \pi_i^{(k)} p_{ij} \tag{2.4}$$

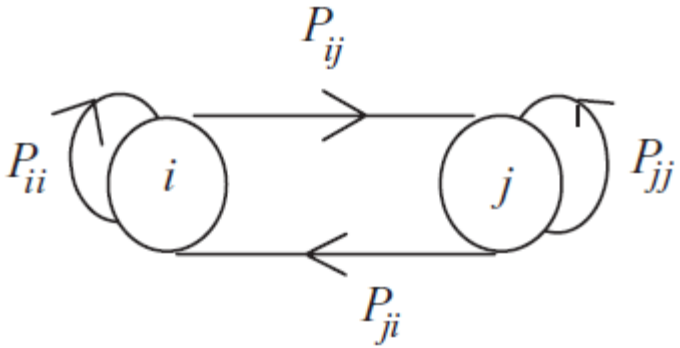


Figure 2.1 State transition diagram

2.1. Defi nition of Discrete-time Markov Chains

Example

A passenger lift in a shopping complex of three storeys is capable of stopping at every fl oor, depending on the passengers' traffi c pattern. If the lift takes one time interval to go from one destination to another regardless of the number of fl oors between them, and the passenger traffi c pattern is as shown in Table 2.1.

Then the position of the lift at the end of some time intervals in the future is clearly a Markov chain as the lift position at the next time step depends completely on its current position. Its state transition diagram is depicted in Figure 2.2.

Let X_k denotes the level at which we find the lift after k transitions and X_0 be the lift's initial position at time 0. $\pi_i^{(k)}$ is the probability of the lift in state i after k transitions.

We are given that the lift is at ground floor level at time 0. It is equivalent to saying that

2.1. Defi nition of Discrete-time Markov Chains

Table 2.1 Passengers' traffi c demand

Lift present position (current state)	Probability of going to the next level		
	ground floor (state 0)	1st floor (state 1)	2nd floor (state 2)
ground floor (state 0)	0	0.5	0.5
1st floor (state 1)	0.75	0	0.25
2nd floor (state 2)	0.75	0.25	0

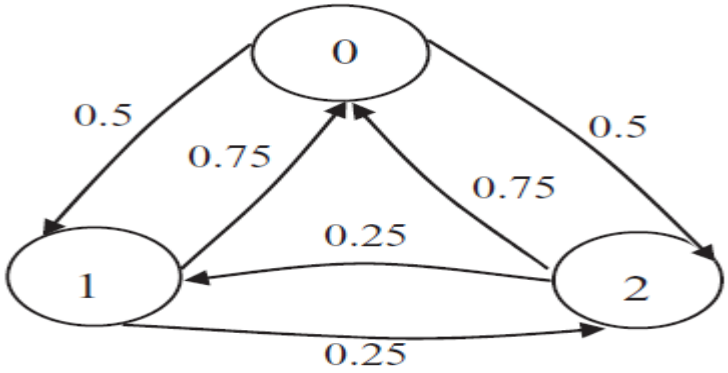


Figure 2.2 State transition diagram for the lift example

2.1. Defi nition of Discrete-time Markov Chains

$$\pi_0^0 \equiv P(X_0 = 0) = 1$$

$$\pi_1^0 \equiv P(X_0 = 1) = 0$$

$$\pi_2^0 \equiv P(X_0 = 2) = 0$$

Or in vector form

$$\tilde{\pi}^{(0)} = [\pi_0^0, \pi_1^0, \pi_2^0] = [1, 0, 0]$$

2.1. Defi nition of Discrete-time Markov Chains

(i) The probability of the lift's position after 1st transition:

$$\pi_0^1 = P(X_1 = 0)$$
$$= \sum_i P(X_1 = 0 | X_0 = i) P(X_0 = i)$$
$$= 0$$
$$\pi_1^1 \equiv P(X_1 = 1)$$
$$= \sum_i P(X_1 = 1 | X_0 = i) P(X_0 = i)$$
$$= 0.5$$

Similarly

$$\pi_2^1 = P(X_1 = 2) = 0.5$$

2.1. Defi nition of Discrete-time Markov Chains

(ii) The probability of the lift's position after 2nd transition:

$$\begin{aligned}\pi_0^2 &\equiv P(X_2 = 0) \\ &= \sum_i P(X_2 = 0 | X_1 = i) P(X_1 = i) \\ &= 0 \times 0 + 0.75 \times 0.5 + 0.75 \times 0.5 \\ &= 0.75\end{aligned}$$

and

$$\pi_1^2 = P(X_2 = 1) = 0.125$$

$$\pi_2^2 = P(X_2 = 2) = 0.125$$

From the above calculations, we see that we need only the probabilities of the lift's position after the 1st transition in order to calculate its probabilities after the 2nd transition. That is to say that the future development of the process depends only on its current position.

2.2. Matrix Formulation of State Probabilities

Students who have refreshed their memory of matrices in Chapter 1 will recognize that the above calculations can be formulated more succinctly in terms of matrix operations. Let us express the transition probabilities in an $n \times n$ square matrix $\tilde{P}$, assuming there are n states. The square matrix is known as *transition probability matrix*, or *transition matrix* for short:

$$\tilde{P} = (p_{ij}) = \begin{pmatrix} p_{11} & p_{12} & \cdots & p_{1n} \\ p_{21} & p_{22} & \cdots & p_{2n} \\ \cdots & & p_{ij} & \\ \cdots & & & p_{nn} \end{pmatrix} \qquad (2.5)$$

2.2. Matrix Formulation of State Probabilities

The element *pij* is the transition probability defi ned in Equation (2.1). If the number of states is fi nite, say *n*, then we have a *n x n* matrix $P^{\sim}$, otherwise the matrix is infi nite. Since the probability of going to all other states, including itself, should sum to unity, as shown in Equation (2.3), the sum of each row in the above matrix should equal to unity, that is:

$$\sum_{j} p_{ij} = 1$$

A matrix with each row sums to unity and all elements positive or zero are called a *stochastic matrix*. A Markov chain is completely characterized by this (one-step) transition probability matrix together with the initial probability vector.

Similarly, we express the state probabilities at each time interval as a row vector:

2.2. Matrix Formulation of State Probabilities

$$\tilde{\pi}^{(k)} = (\pi_0^{(k)}, \pi_1^{(k)}, \ldots, \pi_n^{(k)}) \tag{2.6}$$

Using these matrix notations, can be formulated as

$$\begin{aligned} \tilde{\pi}^{(1)} &= \tilde{\pi}^{(0)}\tilde{P} \\ \tilde{\pi}^{(2)} &= \tilde{\pi}^{(1)}\tilde{P} \\ &\ldots \\ \tilde{\pi}^{(k)} &= \tilde{\pi}^{(k-1)}\tilde{P} \end{aligned} \tag{2.7}$$

Back substituting the p˜(*i*), we have from Equation (2.7) the following equation:

$$\tilde{\pi}^{(k)} = \tilde{\pi}^{(0)}\tilde{P}^{(k)} \tag{2.8}$$

where $$\tilde{P}^{(k)} = \tilde{P}.\tilde{P}^{(k-1)} = \tilde{P}^k \tag{2.9}$$

2.2. Matrix Formulation of State Probabilities

$\tilde{P}^k$, the so-called *k-step* transition matrix, is the k-fold multiplication of the one-step transition matrix by itself. We define $\tilde{P}^{(0)} = I$.

Equations (2.8) and (2.9) give us a general method for calculating the state probability *k* steps into a chain. From matrix operations, we know that

$$\tilde{P}^{(k+l)} = \tilde{P}^{(k)} \times \tilde{P}^{(l)} \tag{2.10}$$

or

$$P_{ij}^{k+l} = \sum_{k=0}^{n} P_{ik}^{k} P_{kj}^{l} \tag{2.11}$$

These two equations are the well-known Chapman–Kolmogorov forward equations.

Example

For the lift example, the transition probability matrix is

$$\tilde{P} = \begin{pmatrix} 0 & 0.5 & 0.5 \\ 0.75 & 0 & 0.25 \\ 0.75 & 0.25 & 0 \end{pmatrix}$$

The transition matrices for the first few transitions can be computed as

$$\tilde{P}^{(2)} = \tilde{P}^{(1)} \times \tilde{P}^{(1)} = \begin{pmatrix} 0.75 & 0.125 & 0.125 \\ 0.1875 & 0.4375 & 0.375 \\ 0.1875 & 0.375 & 0.4375 \end{pmatrix}$$

Example

$$\tilde{P}^{(3)} = \begin{pmatrix} 0.1875 & 0.4063 & 0.4062 \\ 0.6094 & 0.1875 & 0.2031 \\ 0.6094 & 0.2031 & 0.1875 \end{pmatrix}$$

$$\tilde{P}^{(4)} = \begin{pmatrix} 0.6094 & 0.1953 & 0.1953 \\ 0.21930 & 0.3555 & 0.3515 \\ 0.2930 & 0.3515 & 0.3554 \end{pmatrix}$$

Example

If the lift is in state 0 at time 0, i.e. $\tilde{\pi}^{(0)} = (1, 0, 0)$, then we have

$$\tilde{\pi}^{(1)} = \tilde{\pi}^{(0)} \times \tilde{P}^{(1)} = (0, 0.5, 0.5)$$

$$\tilde{\pi}^{(2)} = (0.75, 0.125, 0.125)$$

$$\tilde{\pi}^{(3)} = (0.1875, 0.4062, 0.4063)$$

If $\tilde{\pi}^{(0)} = (0, 1, 0)$, we have

$$\tilde{\pi}^{(1)} = \tilde{\pi}^{(0)} \times \tilde{P}^{(1)} = (0.75, 0, 0.25)$$

$$\tilde{\pi}^{(2)} = (0.1875, 0.4375, 0.375)$$

$$\tilde{\pi}^{(3)} = (0.6095, 0.1875, 0.2031)$$

Example

If $\tilde{\pi}^{(0)} = (0, 0, 1)$, we have

$$\tilde{\pi}^{(1)} = \tilde{\pi}^{(0)} \times \tilde{P}^{(1)} = (0.75, 0.25, 0)$$

$$\tilde{\pi}^{(2)} = (0.1875, 0.3750, 0.4375)$$

$$\tilde{\pi}^{(3)} = (0.6094, 0.2031, 0.1875)$$

The stationary probabilities π_j are uniquely determined through the following equations:

$$\sum_j \pi_j = 1 \qquad (2.12)$$

$$\pi_j = \sum_i \pi_i P_{ij} \qquad (2.13)$$

2.4. Classical m/m/1 queue

This classical M/M/1 queue refers to a queueing system where customers arrive according to a Poisson process and are served by a single server with an exponential service-time distribution. The arrival rate and service rate do not depend upon the number of customers in the system so are state-independent. Recall that the defaults for the other two parameters in Kendall's notation are infi nite system capacity and fi rst-come fi rst-served queueing discipline.

Firstly, let us focus our attention on the number of customers $N(t)$ in the system at time t. A customer arriving at the system can be considered as a birth and a customer leaving the system after receiving his service is deemed as a death. Since the Poisson process prohibits the possibility of having more than one arrival in Δt and the exponential service time ensures that there is at most one departure in Δt, then clearly $N(t)$ is a birth-death process because it can only go to its neighbouring states, $(N(t) + 1)$ or $(N(t) - 1)$ in a time interval Δt.

2.4. Classical m/m/1 queue

To evaluate P_0, we use the normalization condition $\sum_{k=0}^{\infty} p_k = 1$

we have $$P_0 = 1 - \frac{\lambda}{\mu} = 1 - \rho \qquad (2.14)$$

and $$P_k = (1-\rho)\rho^k \qquad (2.15)$$

We see that the probability of having k customers in the system is a geometric random variable with parameter ρ. It should be noted from the expression that ρ has to be less than unity in order for the sequence to converge and hence be a stable system.

2.4.1. Global and local balance concepts

Before we proceed, let us deviate from our discussion of P_k and examine a very simple but powerful concept, *global balance* of probability flows. Recall the equilibrium stationary equation we derived in the preceding section for M/M/1:

$$(\lambda + \mu)P_k = \lambda P_{k-1} + \mu P_{k+1} \qquad (2.16)$$

We know from probability theory that P_{k-1} is the fraction of time that the process found in state $(k-1)$, therefore λP_{k-1} can be interpreted as the expected rate of transitions from state $(k-1)$ to state k and this quantity is called the *stochastic* (or *probability*) *flow* from state $(k-1)$ to state k.

2.4.1. Global and local balance concepts

Similarly, μP_{k+1} is the stochastic fl ow going from state $(k + 1)$ to state k. Thus, we see that the right-hand side of the equation represents the total stochastic fl ow into state k and the left-hand side represents the stochastic fl ow out of state k. In other words, Equation (2.16) tells us that the total stochastic fl ow in and out of a state should be equal under equilibrium condition, as shown in Figure 2.3. Equation (2.16) above is called the *global balance equation* for the Markov chain in question:

2.4.1. Global and local balance concepts

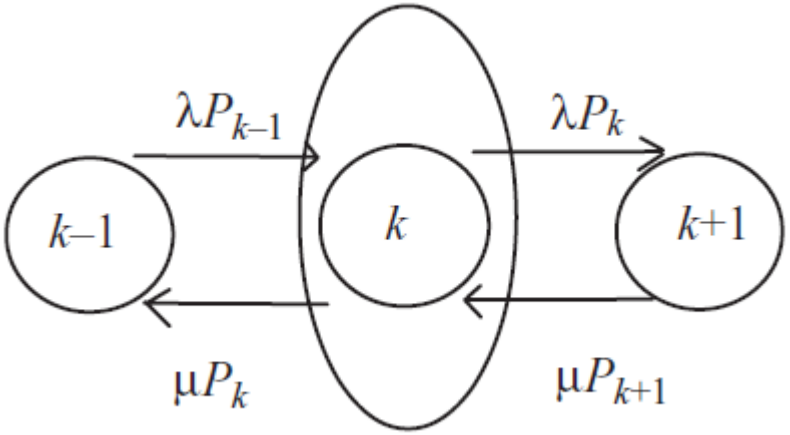


Figure 2.3 Global balance concept

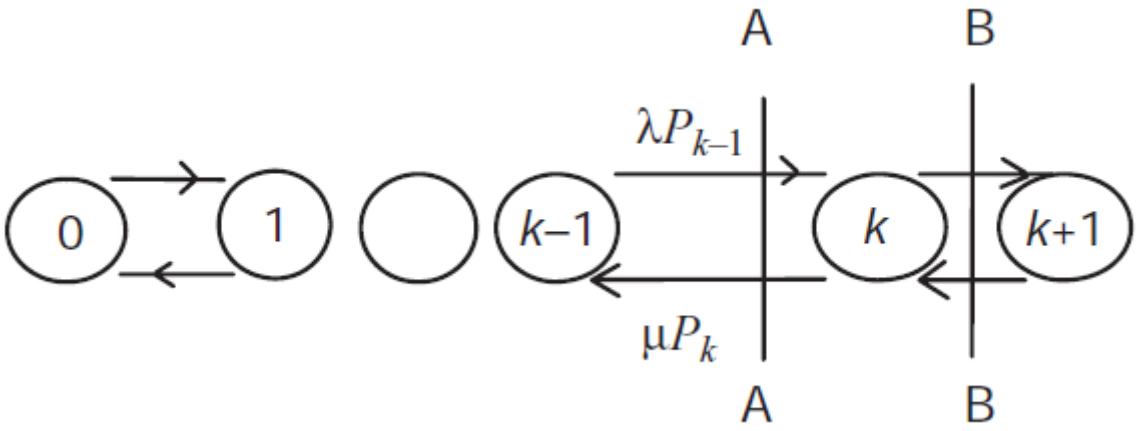


Figure 2.4 Local balance concept

2.4.1. Global and local balance concepts

Extending our discussion of fl ow balancing further, let us consider an imaginary boundary B–B between node k and $k + 1$, as shown in Figure 2.4. If we equate the probability fl ows that go across this boundary, we have

$$\lambda P_k = \mu P_{k+1} \tag{2.17}$$

This equation can be interpreted as a special case of the global balance in the sense that it equates the flow in and out of an imaginary super node that encompasses all nodes from node 0 to node k. This particular expression is referred as the *local balance equation* or *detailed balance equation*. It can be verified that this local balance equation always satisfies the global balance equation.

2.4.2. Performance measures of the m/m/1 system

Coming back to our discussion of P_k and using the local balance equation across the boundary A–A of node $(k-1)$ and node k, we have

$$P_k = \frac{\lambda}{\mu} P_{k-1} = \left(\frac{\lambda}{\mu}\right)^2 P_{k-2} = \left(\frac{\lambda}{\mu}\right)^3 P_{k-3} \qquad (2.18)$$
$$= \rho^k P_0$$

As usual, we compute P_0 using the normalization equation $\sum_k P_k = 1$ to give

$$P_0 = (1-\rho) \qquad (2.19)$$

$$P_k = (1-\rho)\rho^k \qquad (2.20)$$

2.4.2. Performance measures of the m/m/1 system

(i) The probability of having n or more customers in the system is given by

$$
\begin{aligned}
P[N \geq n] &= \sum_{k=n}^{\infty} P_k \\
&= (1-\rho)\sum_{k=n}^{\infty} \rho^k \\
&= (1-\rho)\left[\sum_{k=0}^{\infty} \rho^k - \sum_{k=0}^{n-1} \rho^k\right] \\
&= (1-\rho)\left[\frac{1}{1-\rho} - \frac{1-\rho^n}{1-\rho}\right] \\
&= \rho^n
\end{aligned}
\qquad (2.21)
$$

2.4.2. Performance measures of the m/m/1 system

(ii) The average number of customers N in the system in steady state is then given by

$$N = \frac{\rho}{1-\rho} = \frac{\lambda}{\mu - \lambda} \tag{2.22}$$

(iii) The average time (T) a customer spends in the system is sometimes referred to as system time, system delay or queueing time in other queueing literature. We may use them interchangeably:

$$T = \frac{N}{\lambda} = \frac{\rho}{\lambda(1-\rho)}$$

$$= \frac{1}{\mu - \lambda} \tag{2.23}$$

2.4.2. Performance measures of the m/m/1 system

(iv) The average customers at the service facility N_s:

$$N_s = \lambda/\mu = \rho \\ = 1 - P_0 \qquad (2.24)$$

(V) The average time a customer spends in the waiting queue is also known as *waiting time*. Students should not confuse this with the queueing time, which is the sum of waiting time and service time:

$$W = T - \frac{1}{\mu} \\ = \frac{\rho}{\mu - \lambda} \qquad (2.25)$$

2.4.2. Performance measures of the m/m/1 system

(vi) The average number of customers in the waiting queue:

$$N_q = \lambda W$$

$$= \frac{\rho^2}{1-\rho} \qquad (2.26)$$

2.5. M/M/1 system time (delay) distribution

In previous sections, we obtained several performance measures of an M/M/1 queue, so we are now in a position to predict the long-range averages of this queue. However, we are still unable to say anything about the probability that a customer will spend up to 3 minutes in the system or answer any questions on that aspect. To address them, we need to examine the actual probability distribution of both the system time and waiting time. Let us define the system-time density function as

$$f_T(t) = \frac{d}{dt} P[T < t] \qquad (2.27)$$

Students should note that the probability distributions (or density functions) of system time or waiting time depend very much on the actual service discipline used, although the steady state performance measures are independent of the service discipline.

2.5. M/M/1 system time (delay) distribution

We assume here that FCFS discipline is used and focus our attention on the arrival of customer i. The system time of this customer will be the sum of his/her service time and the service times of those customers (say k of them) ahead of him/her if the system is not empty when he/she arrives. Otherwise, his/her system time will be just his/her own service time. That is

$$T = \begin{cases} x_i + x_1 + x_2 + \ldots + x_k & k \geq 1 \\ x_i & k = 0 \end{cases} \qquad (2.28)$$

and the cumulative distribution function $F_T(t)$ is

$$F_T(t) = P[T \leq t] = 1 - e^{-(\mu-\lambda)t} \qquad (2.29)$$

2.5. M/M/1 system time (delay) distribution

Integrating we have the waiting time cumulative distribution function $F_w(t)$:

$$F_w(t) = 1 - \rho e^{-(\mu-\lambda)t} \qquad (2.30)$$

2.5. M/M/1 system time (delay) distribution

Example

At a neighhourhood polyclinic, patients arrive according to a Poisson process with an average inter-arrival time of 18 minutes. These patients are given a queue number upon arrival and will be seen, by the only doctor manning the clinic, according to their queue number. The length of a typical consultation session is found from historical data to be exponentially distributed with a mean of 7 minutes:

(i) What is the probability that a patient has to wait to see the doctor?
(ii) What is the average number of patients waiting to see the doctor?
(iii) What is the probability that there are more than 5 patients in the clinic, including the one in consultation with the doctor?
(iv) What is the probability that a patient would have to wait more than 10 minutes for this consultation?
(v) The polyclinic will employ an additional doctor if the average waiting time of a patient is at least 7 minutes before seeing the doctor. By how much must the arrival rate increase in order to justify the additional doctor?

Solution

Assuming the clinic is sufficiently large to accommodate a large number of patients, the situation can be modelled as an M/M/1 queue. Given the following parameters:

$$\lambda = 1/18 \text{ person/min} \qquad \text{and} \qquad \mu = 1/7 \text{ person/min}$$

We have $\rho = 7/18$:

(i) $P[\text{patient has to wait}] = \rho = 1 - P_0 = 7/18$

Solution

Assuming the clinic is sufficiently large to accommodate a large number of patients, the situation can be modelled as an M/M/1 queue. Given the following parameters:

$$\lambda = 1/18 \text{ person/min} \qquad \text{and} \qquad \mu = 1/7 \text{ person/min}$$

We have $\rho = 7/18$:

(ii) The waiting-queue length $N_q = \rho^2 /(1 - \rho)$

Solution

Assuming the clinic is sufficiently large to accommodate a large number of patients, the situation can be modelled as an M/M/1 queue. Given the following parameters:

$$\lambda = 1/18 \text{ person/min} \qquad \text{and} \qquad \mu = 1/7 \text{ person/min}$$

We have $\rho = 7/18$:

(ii) The waiting-queue length $N_q = \rho^2 /(1 - \rho) = 49/198$

Solution

Assuming the clinic is sufficiently large to accommodate a large number of patients, the situation can be modelled as an M/M/1 queue. Given the following parameters:

$$\lambda = 1/18 \text{ person/min} \qquad \text{and} \qquad \mu = 1/7 \text{ person/min}$$

We have $\rho = 7/18$:

(iii) $P[N \geq 5] = \rho^5$

Solution

Assuming the clinic is sufficiently large to accommodate a large number of patients, the situation can be modelled as an M/M/1 queue. Given the following parameters:

$$\lambda = 1/18 \text{ person/min} \qquad \text{and} \qquad \mu = 1/7 \text{ person/min}$$

We have $\rho = 7/18$:

(iii) $P[N \geq 5] = \rho^5 = 0.0089$

Solution

Assuming the clinic is sufficiently large to accommodate a large number of patients, the situation can be modelled as an M/M/1 queue. Given the following parameters:

$$\lambda = 1/18 \text{ person/min} \qquad \text{and} \qquad \mu = 1/7 \text{ person/min}$$

We have $\rho = 7/18$:

(iv) Since $P[waitingtime \leq 10] = 1 - \rho e^{-\mu(1-\rho)t}$.

Solution

Assuming the clinic is sufficiently large to accommodate a large number of patients, the situation can be modelled as an M/M/1 queue. Given the following parameters:

$$\lambda = 1/18 \text{ person/min} \qquad \text{and} \qquad \mu = 1/7 \text{ person/min}$$

We have $\rho = 7/18$:

(iv) Since $P[waitingtime \leq 10] = 1 - \rho e^{-\mu(1-\rho)t}$,

$$P[\text{queueing time} > 10] = \frac{7}{18} e^{-1/7(1-7/18)\times 10} = 0.162$$

Solution

Assuming the clinic is sufficiently large to accommodate a large number of patients, the situation can be modelled as an M/M/1 queue. Given the following parameters:

$$\lambda = 1/18 \text{ person/min} \qquad \text{and} \qquad \mu = 1/7 \text{ person/min}$$

We have $\rho = 7/18$:

(v) Let the new arrival rate be λ'

$$\text{then } \frac{7\lambda'}{\frac{1}{7} - \lambda'} \geq 7$$

Solution

Assuming the clinic is sufficiently large to accommodate a large number of patients, the situation can be modelled as an M/M/1 queue. Given the following parameters:

$$\lambda = 1/18 \text{ person/min} \qquad \text{and} \qquad \mu = 1/7 \text{ person/min}$$

We have $\rho = 7/18$:

(v) Let the new arrival rate be λ'

then $\dfrac{7\lambda'}{\dfrac{1}{7} - \lambda'} \geq 7 \Rightarrow \lambda' = \dfrac{1}{14}$ person/min person/min.

2.5. M/M/1/S queueing systems

The M/M/1 model discussed earlier is simple and useful if we just want to have a first-cut estimation of a system's performance. However, it becomes a bit unrealistic when it is applied to real-life problems, as most of them do have physical capacity constraints. Often we have a finite waiting queue instead of one that can accommodate an infinite number of customers. The M/M/1/S that we shall discuss is a more accurate model for this type of problem.

In *M/M/1/S*, the system can accommodate only *S* customers, including the one being served. Customers who arrive when the waiting queue is full are not allowed to enter and have to leave without being served. The state transition diagram is the same as the classical M/M/1 queue except that it is truncated at state *S*, as shown in Figure 2.3. This truncation of state transition diagram will affect the queueing results through *P*0.

2.5. M/M/1/S queueing systems

From last section, we have

$$P_k = \rho^k P_0 \qquad \text{where} \qquad \rho = \lambda/\mu$$

Using the normalization equation but sums to S state, we have

$$P_0 \sum_{k=0}^{S} \rho^k = 1 \quad \Rightarrow \quad P_0 = \frac{1-\rho}{1-\rho^{S+1}} \tag{2.31}$$

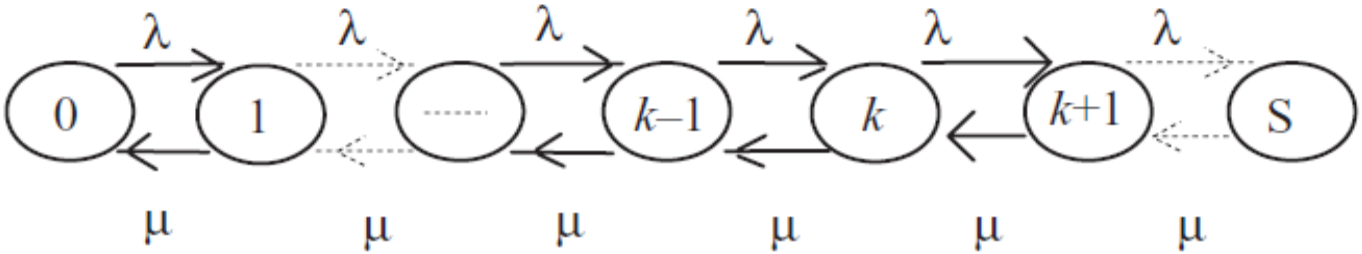


Figure 2.3 M/M/1/S transition diagram

2.5. M/M/1/S queueing systems

$$P_k = \frac{(1-\rho)\rho^k}{1-\rho^{S+1}} \tag{2.32}$$

It should be noted that this system will always be ergodic and an equilibrium condition exists, even for the case where $\lambda \geq \mu$. This is due to the fact that the system has a self-regulating mechanism of turning away customers and hence the queue cannot grow to infinity. Students should note that the effective arrival rate that goes into the system is always less than μ.

2.5.1. Blocking Probability

Let us now digress from our discussion and look at the concept of *blocking probability Pb*. This is the probability that customers are blocked and not accepted by the queueing system because the system capacity is full. This situation occurs in queueing systems that have a fi nite or no waiting queue, hence it does not need to be an M/M/1/S queue. It can be any queueing system that blocks customers on arrival (Figure 2.4).

When the waiting queue of a system is full, arriving customers are blocked and turned away. Hence, the arrival process is effectively being split into two Poisson processes probabilistically through the blocking probability *Pb*. One stream of customers enters the system and the other is turned away, as shown in Figure 2.4.

The net arrival rate is

$$\lambda' = \lambda(1 - P_b)$$

2.5.1. Blocking Probability

For a stable system, the net departure rate γ should be equal to the net arrival rate, otherwise the customers in the system either will increase without bound or simply come from nowhere. Therefore

$$\gamma = \lambda(1 - P_b)$$

However, we know that the net departure rate can be evaluated by

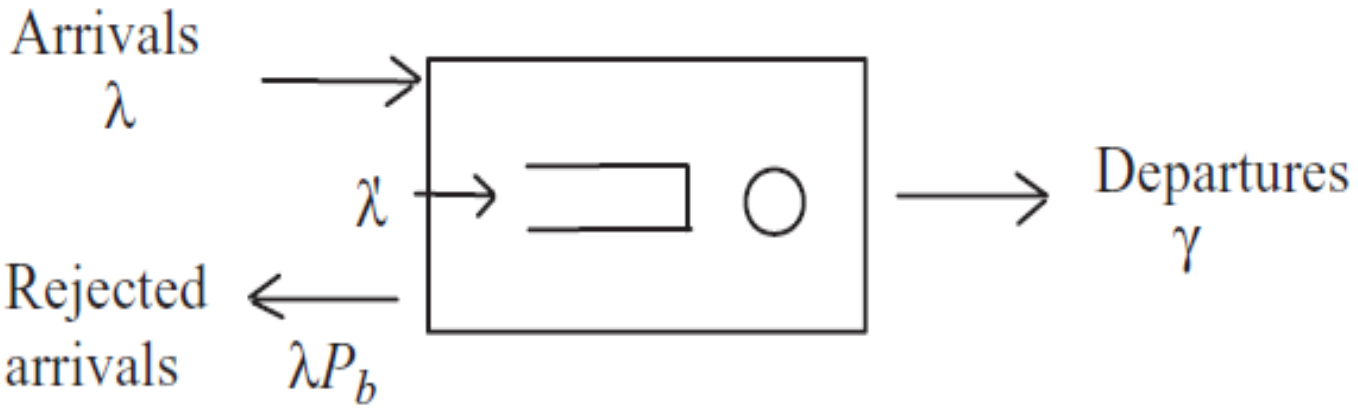


Figure 2.4 Blocking probability of a queueing

2.5.1. Blocking Probability

$$\gamma = \sum_{k=1}^{S} \mu P_k = \mu(1 - P_0) \tag{2.33}$$

Equating both expressions, we obtain

$$\lambda(1 - P_b) = \mu(1 - P_0)$$

$$P_b = \frac{P_0 + \rho - 1}{\rho} \tag{2.34}$$

The expression is derived for a queueing system with constant arrival and service rates

2.5.2. Performance Measures of M/M/1/S Systems

Continuing with our early discussion, the various performance measures can then be computed once P_k is found:

(i) The saturation probability

By definition, the saturation probability is the probability when the system is full; that is there are S customers in the system. We say the system is saturated and we have

$$P_S = \frac{(1-\rho)\rho^S}{1-\rho^{S+1}}$$

However, substituting *P*0 of M/M/1/S into expression (2.33), we have

$$P_b = \frac{P_0 + \rho - 1}{\rho} = \frac{(1-\rho)\rho^S}{1-\rho^{S+1}} \qquad (2.35)$$

2.5.2. Performance Measures of M/M/1/S Systems

That is just the expression for P_s. This result is intuitively correct as it indicates that the blocking probability is the probability when the system is saturated. Owing to this blocking probability, the system now has a built-in self-regulatory mechanism and we see for the first time that it is still stable even when arrival rate is great than the service rate.

For $\rho = 1$, the expression for the blocking probability has to be evaluated using L'Hopital's rule and we have

$$P_b = \frac{1}{S+1}$$

2.5.2. Performance Measures of M/M/1/S Systems

(ii) The average number of customers in the system is given by

$$
\begin{aligned}
N &= \sum_{k=0}^{S} kP_k \\
&= \sum_{k=1}^{S} \left(\frac{1-\rho}{1-\rho^{S+1}} \right) k\rho^k \\
&= \left(\frac{1-\rho}{1-\rho^{S+1}} \right) \rho \sum_{k=1}^{s} k\rho^{k-1} \\
&= \left(\frac{1-\rho}{1-\rho^{S+1}} \right) \rho \frac{d}{d\rho} \sum_{k=1}^{s} \rho^k \\
&= \frac{\rho}{1-\rho} - \frac{(S+1)\rho^{S+1}}{1-\rho^{S+1}} \\
&= \frac{\rho}{1-\rho} - \frac{\rho}{1-\rho}(S+1)P_S
\end{aligned}
\qquad (2.36)
$$

2.5.2. Performance Measures of M/M/1/S Systems

Again, for $\rho = 1$, the expression has to be evaluated by L'Hopital's rule and we have

$$N = \frac{S}{2} \tag{2.37}$$

(iii) The average number of customers at the service facility:

$$\begin{aligned} N_s &= P[k=0]E[N_s|k=0] + P[k>0]E[N_s|N>0] \\ &= 1 - P_0 \\ &= \rho(1 - P_S) \end{aligned} \tag{2.38}$$

(iv) The average number of customers in the waiting queue:

$$N_q = N - N_s = \frac{\rho^2}{1-\rho} - \frac{(S+\rho)\rho}{1-\rho} P_S \tag{2.39}$$

2.5.2. Performance Measures of M/M/1/S Systems

(v) The average time spent in the system and in the waiting queue.
Since customers are blocked when there are S customers in the system, the effective arrival rate of customers admitted into the system is

$$\lambda' = \lambda(1 - P_s) \tag{2.40}$$

2.5.2. Performance Measures of M/M/1/S Systems

and the T and W can be computed as

$$T = \frac{N}{\lambda'} = \frac{1}{\mu - \lambda} - \frac{S\rho^{S+1}}{\lambda - \mu\rho^{S+1}} \tag{2.41}$$

$$W = \frac{N_q}{\lambda'} = \frac{\rho}{\mu - \lambda} - \frac{S\rho^{S+1}}{\lambda - \mu\rho^{S+1}} \tag{2.42}$$

2.6. MULTI-SERVER SYSTEMS – M/M/m

Having examined the classical single-server queueing system, it is natural for us now to look at its logical extension; the multi-server queueing system in which the service facility consists of m identical parallel servers, as shown in Figure 2.5. Here, identical parallel servers mean that they all perform the same functions and a customer at the head of the waiting queue can go to any of the servers for service.

If we again focus our attention on the system state $N(t)$, then $\{N(t), t \geq 0\}$ is a birth-death process with state-dependent service rates. When there is one customer, one server is engaged in providing service and the service rate is μ. If there are two customers, then two servers are engaged and the total service rate is 2μ, so the service rate increases until $m\mu$ and stays constant thereafter.

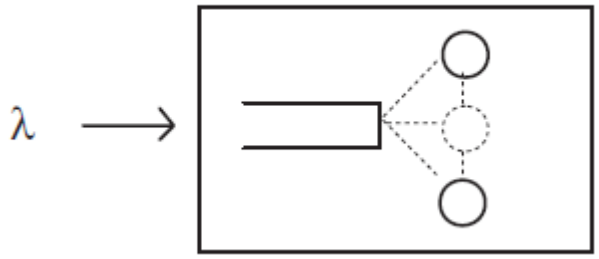


Figure 2.5 A multi-server system model

2.6. MULTI-SERVER SYSTEMS – M/M/m

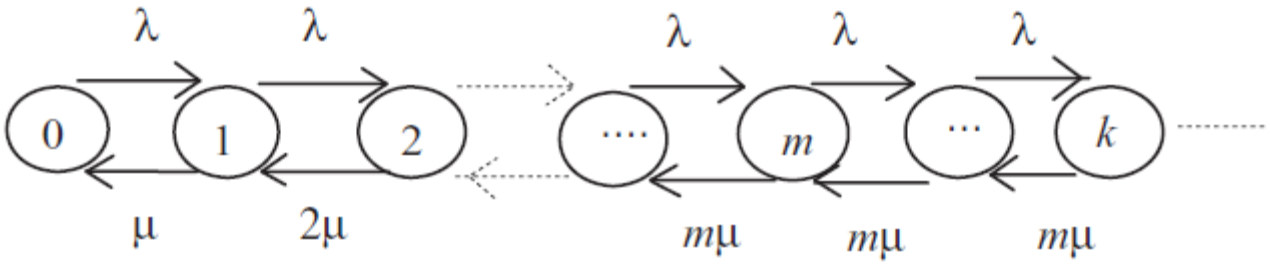


Figure 2. 6 M/M/m transition diagram

There are several variations of multi-server systems. We shall fi rst examine the *M/M/m* with an infi nite waiting queue. Its corresponding state transition diagram is shown in Figure 2. 6.

Using local balance concept, we can readily write down the governing equations by inspection:

$$k \le m \quad k\mu P_k = \lambda P_{k-1} \tag{2.43}$$

$$k \ge m \quad m\mu P_k = \lambda P_{k-1} \tag{2.44}$$

2.6. MULTI-SERVER SYSTEMS – M/M/m

We defi ned the utilization as $\mathbf{p = \lambda / m\mu}$ for a multiserver system. Hence, we have from Equation (2.43):

$$
\begin{aligned}
P_k &= \frac{\lambda}{k\mu} P_{k-1} \\
&= \left(\frac{\lambda}{k\mu}\right)\left(\frac{\lambda}{(k-1)\mu}\right) P_{k-2} \\
&= \frac{(m\rho)^k}{k!} P_0
\end{aligned}
$$

2.6. MULTI-SERVER SYSTEMS – M/M/m

From Equation (2.44) we have:

$$P_k = \frac{\lambda}{m\mu} P_{k-1},$$

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

$$= \frac{(m\rho)^{k-m}}{m^{k-m}} \left(\frac{m^m \rho^m}{m!} P_0\right)$$

$$= \frac{m^m \rho^k}{m!} P_0$$

Hence

$$P_k = \begin{cases} P_0 \dfrac{(m\rho)^k}{k!} & k \le m \\ P_0 \dfrac{m^m \rho^k}{m!} & k \ge m \end{cases} \qquad (2.45)$$

2.6. MULTI-SERVER SYSTEMS – M/M/m

Using the normalization condition $\sum P_k = 1$, we obtain

$$P_0\left[\sum_{k=0}^{m-1}\frac{(m\rho)^k}{k!}+\sum_{k=m}^{\infty}\frac{m^m\rho^k}{m!}\right]=1$$

$$P_0=\left[\sum_{k=0}^{m-1}\frac{(m\rho)^k}{k!}+\frac{(m\rho)^{\boldsymbol{m}}}{m!}\sum_{k=m}^{\infty}\rho^{k-m}\right]^{-1}$$

$$=\left[\sum_{k=0}^{m-1}\frac{(m\rho)^k}{k!}+\frac{(m\rho)^m}{m!(1-\rho)}\right]^{-1} \qquad (2.46)$$

2.6.1. Performance Measures of M/M/m Systems

(i) The probability of delay

This is the probability that an arriving customer finds all servers busy and is forced to wait in the queue. This situation occurs when there are more than m customers in the system, hence we have

$$P_d = P[delay] = \sum_{k=m}^{\infty} P_k$$

$$= \frac{P_o(m\rho)^m}{m!} \sum_{k=m}^{\infty} \rho^{k-m}$$

$$= \frac{P_0(m\rho)^m}{m!(1-\rho)} \qquad (2.47)$$

2.6.1. Performance Measures of M/M/m Systems

(ii) As the probability mass function P_k consists of two functions, it is easier to first find N_q, the number of customers waiting in the queue, instead of N so that the discontinuity in *pmf* can be avoided:

$$N_q = \sum_{k=0}^{\infty} kP_{m+k} = P_0 \frac{(m\rho)^m}{m!} \sum_{k=0}^{\infty} k\rho^k$$

$$= P_0 \frac{(m\rho)^m}{m!} \frac{\rho}{(1-\rho)^2}$$

$$= \frac{\rho}{1-\rho} P_d = \frac{\lambda}{m\mu - \lambda} P_d \qquad (2.48)$$

2.6.1. Performance Measures of M/M/m Systems

(iii) The time spent in the waiting queue:

$$W = \frac{N_q}{\lambda} = \frac{1}{m\mu - \lambda} P_d \qquad (2.49)$$

(iv) The time spends in the queueing system:

$$T = W + \frac{1}{\mu} = \frac{P_d}{m\mu - \lambda} + \frac{1}{\mu} \qquad (2.50)$$

(v) The number of customers in the queueing system:

$$N = \lambda T = \frac{\rho}{1-\rho} P_d + m\rho \qquad (2.51)$$

2.6.1. Performance Measures of M/M/m Systems

Example

At the telephone hot line of a travel agency, information enquiries arrive according to a Poisson process and are served by 3 tour coordinators. These tour coordinators take an average of about 6 minutes to answer an enquiry from each potential customer.

From past experience, 9 calls are likely to be received in 1 hour in a typical day. The duration of these enquiries is approximately exponential. How long will a customer be expected to wait before talking to a tour coordinator, assuming that customers will hold on to their calls when all coordinator are busy? On the average, how many customers have to wait for these coordinators?

Solution

The situation can be modelled as an M/M/3 queue with

$$\rho = \frac{\lambda}{m\mu} = \frac{9/60}{3 \times (1/6)} = 0.3$$

2.6.1. Performance Measures of M/M/m Systems

$$P_0 = \left[\sum_{k=0}^{m-1}\frac{(m\rho)^k}{k!} + \frac{(m\rho)^m}{m!}\left(\frac{1}{1-\rho}\right)\right]^{-1}$$

$$= \left[\sum_{k=0}^{2}\frac{(0.9)^k}{k!} + \frac{(0.9)^3}{3!}\left(\frac{1}{1-0.3}\right)\right]^{-1}$$

$$= 0.4035$$

$$P_d = P_0\frac{(m\rho)^m}{m!}\left(\frac{1}{1-\rho}\right)$$

$$= 0.4035 \times \frac{(0.9)^3}{3!} \times \frac{1}{1-0.3}$$

$$= 0.07$$

$$N_q = \frac{\rho}{1-\rho}P_d = 0.03$$

$$W = \frac{N_q}{\lambda} = \frac{0.03}{9/60} = 0.2 \text{ minute}$$

2.7 . PRIORITY QUEUEING SYSTEMS

For all the queueing systems that we have discussed so far, the arriving customers are treated equally and served in the order they arrive at the system; that is FCFS queueing discipline is assumed. However, in real life we often encounter situations where certain customers are more important than others. They are given greater privileges and receive their services before others. The queueing system that models this kind of situation is called a *priority queueing system*.

There are various applications of this priority model in data networks. In a packet switching network, control packets that carry vital instructions for network operations are usually transmitted with a higher priority over that of data packets. In a multi-media system in which voice and data are carried in the same network, the voice packets may again accord a higher priority than that of the data packets owing to real-time requirements.

2.7 . PRIORITY QUEUEING SYSTEMS

For our subsequent treatments of priority queueing systems, we divide the arriving customers into n different priority classes. The smaller the priority class number, the higher the priority; i.e. Class 1 has the highest priority and Class 2 has the second highest and so on.

There are two basic queueing disciplines for priority systems, namely *pre-emptive* and *non-preemptive*. In a *pre-emptive priority queueing system*, the service of a customer is interrupted when a customer of a higher priority class arrives. As a further division, if the customer whose service was interrupted resumes service from the point of interruption once all customers of higher priority have been served, it is a *pre-emptive resume* system. If the customer repeats his/her entire service, then it is a *pre-emptive repeat* system. In the case of the non-preemptive priority system, a customer's service is never interrupted, even if a customer of higher priority arrives in the meantime.

2.7 . 1. M/G/1 Non- preemptive Priority Queueing

We shall begin with the analysis of an M/G/1 non-preemptive system. In this model, customers of each priority Class i ($i = 1, 2, \ldots n$) arrive according to a Poisson process with rate λ_i and are served by the same server with a general service time distribution of mean $\overline{x_i}$ and second moment $\overline{x_i^2}$ for customers of each Class i, as shown in Figure 2.7 . The arrival process of each class is assumed to be independent of each other and the service process. Within each class, customers are served on their order of arrival. Again the queue for each class of customers is infi nite.

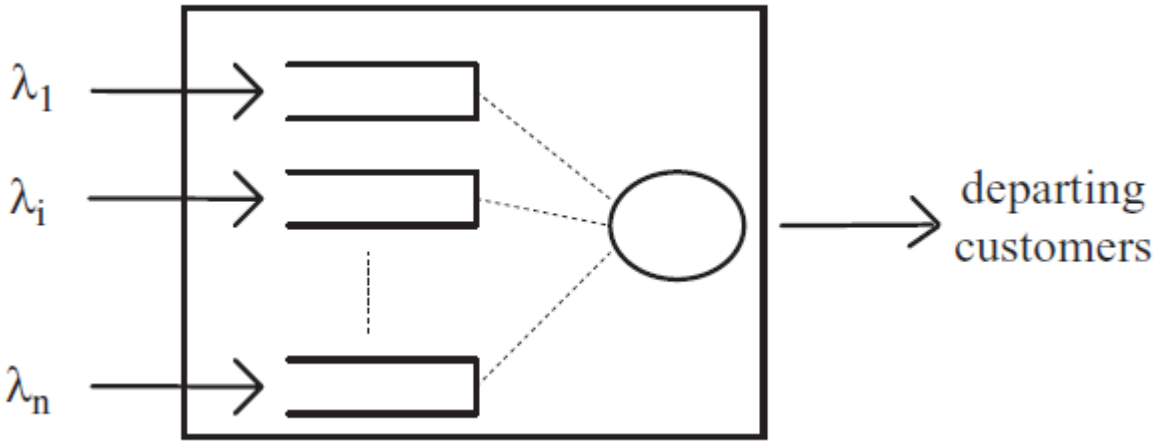


Figure 2.7 M/G /1 nonpreemptive priority system

2.7 . 1. M/G/1 Non- preemptive Priority Queueing

If we define the total arrival $\lambda = \lambda_1 + \lambda_2 + \ldots + \lambda_n$ and utilization of each class of customers $\rho_i = \lambda_i \overline{x_i}$, then we have the average service time $\bar{x}$ and system utilization ρ given by

$$\bar{x} = \frac{1}{\mu} = \frac{\lambda_1}{\lambda}\overline{x_1} + \frac{\lambda_2}{\lambda}\overline{x_2} + \ldots + \frac{\lambda_n}{\lambda}\overline{x_n} \qquad (2.52)$$

$$\rho = \frac{\lambda}{\mu} = \rho_1 + \rho_2 + \ldots + \rho_n \qquad (2.53)$$

The system will reach equilibrium if $\Sigma\rho_i < 1$. However, if this condition is violated then at least some priority classes will not reach equilibrium.

Now let us look at a 'typical' customer C_n of Class i who arrives at the system. His/her mean waiting time in the queue is made up of the following four components:

2.7 . 1. M/G/1 Non- preemptive Priority Queueing

(i) The mean residual service time R for all customers in the system. When a Class i customer arrives, the probability that he/she finds a Class j customer in service is $\rho_j = \lambda_j \overline{x_j}$, therefore R is given by the weighted sum of the residual service time of each class.

$$R = \sum_{k=1}^{n} \rho_k \left(\frac{\overline{x_k^2}}{2\overline{x_k}} \right) = \frac{1}{2} \sum_{k=1}^{n} \lambda_k \overline{x_k^2} \qquad (2.54)$$

(ii) The mean total service time of those customers of the same class ahead of him/her in the waiting queue, that is $\overline{x_i} N_q^i$, where N_q^i is the average number of customers of Class i in the waiting queue.

(iii) The mean total service time of those customers of Class $j (j < i)$ found in the system at the time of arrival; i.e.:

$$\sum_{j=1}^{i-1} \overline{x_j} N_q^j.$$

2.7 . 1. M/G/1 Non- preemptive Priority Queueing

(iv) The mean total service time of those customers of Class $j(j < i)$ arriving at the system while customer C_n is waiting in the queue, i.e.:

$$\sum_{j=1}^{i-1} \overline{x_j} \lambda_j W_i$$

Combining the four components together, we arrive at

$$W_i = R + \overline{x_i} N_q^i + \sum_{j=1}^{i-1} \overline{x_j} N_q^j + \sum_{j=1}^{i-1} \overline{x_j} \lambda_j W_i \qquad (2.55)$$

For Class 1 customers, since their waiting times are not affected by customers of lower classes, the expression of W_1 is the same as that of an M/G/1 queue and is given by

$$W_1 = \frac{R}{1 - \rho_1} \qquad (2.56)$$

2.7 . 1. M/G/1 Non- preemptive Priority Queueing

We can obtain the mean waiting time for class 2 customers as

$$W_2 = \frac{R}{(1-\rho_1)(1-\rho_1-\rho_2)} \qquad (2.57)$$

In general, the expression for the mean waiting time for Class i customers can be calculated recursively using the preceding approach and it yields

$$W_i = \frac{R}{(1-\rho_1-\rho_2-\ldots\rho_{i-1})(1-\rho_1-\rho_2-\ldots\rho_i)} \qquad (2.58)$$

2.7 . 2. Performance Measures of Non- preemptive Priority

The other performance measures can be found once the waiting time is known:

(i) The average number of customers of each class in their own waiting queue:

$$(N_q)_i = \lambda_i W_i = \frac{\lambda_i R}{(1-\rho_1 - \ldots - \rho_{i-1})(1-\rho_1 - \ldots - \rho_{\mathbf{i}})} \qquad (2.59)$$

(ii) The total time spends in the system by a customer of Class *i*:

$$T_i = W_i + \overline{x_i} \qquad (2.60)$$

(iii) Total number of customers in the system:

$$N = \sum_{k=1}^{n} (N_q)_k + \rho \qquad (2.61)$$

2.7 . 2. Performance Measures of Non- preemptive Priority

(iv) If the service times of each class of customers are exponentially distributed with a mean of μ, then in effect we have an M/M/1 non-preemptive priority system. Then we have

$$R = \frac{1}{2}\sum_{k=1}^{n}\lambda_k\left(\frac{1}{\mu}\right)^2 = \frac{\rho}{\mu} \qquad (2.62)$$

The avegare waiting time for classes in system (without the priority scheme)

$$W = \frac{\lambda\overline{x^2}}{2(1-\rho)} \qquad (2.63)$$

2.7 . 3. M/G/1 Preemptive Resume Priority Queueing

This model is the same as the previous one except that now a customer in service can be pre-empted by customers of higher priority. The interrupted service resumes from the point of interruption when all customers of higher priority classes have been served. There are two basic distinctions between this mode and the previous one:

(i) The presence of customers of lower priority Classes ($i + 1$ *to* n) in the system has no effect on the waiting time of a customer of Class i because he/she always pre-empts those customers of lower priority classes. So in the analysis, we can ignore those customers of Class $i + 1$ to n.

(ii) While a customer of Class i is waiting for his/her service, his/her waiting time is the same whether customers of Class 1 to $i - 1$ are served in a pre-emptive manner or non-preemptive fashion. This is due to the fact that he/she only gets his/her service when there are no customers of higher priority classes in the system.

2.7 . 3. M/G/1 Preemptive Resume Priority Queueing

We can use the expression (2.55) for his/her waiting time. Thus, the waiting time in queue of a customer of Class i is given by

$$W_i = \frac{R_i}{(1-\rho_1-\ldots-\rho_{i-1})(1-\rho_1-\ldots-\rho_i)} \quad \text{and} \quad R_i = \frac{1}{2}\sum_{k=1}^{i}\lambda_k \overline{x_k^2} \qquad (2.63)$$

However, the system time of a customer of Class i is not equal to W_i plus his/her service time because his/her service may be interrupted by customers of higher Classes (1 *to* $i-1$) arriving while he/she is being served. Let us define T'_k to be the time his/her service starts until completion, then we have the following expression:

$$T'_i = \overline{x_i} + \sum_{j=1}^{i-1}(\overline{x}_j)\lambda_j T'_i \qquad (2.64)$$

2.7 . 3. M/G/1 Preemptive Resume Priority Queueing

where $\lambda_j T_i'$ is the average arrival of customers of Class j ($j = 1$ *to* $i - 1$) during the time T_i'. Combining these two parts, the system time of a customer of Class i is then given by

$$T_i = W_i + T_i'$$

$$T_i = \frac{R_i}{(1-\rho_i-\ldots-\rho_{i-1})(1-\rho_1-\ldots-\rho_i)} + \frac{\overline{x_i}}{(1-\rho_1-\ldots-\rho_{i-1})} \qquad (2.65)$$

and we arrive at the following expression:

$$T_i = \frac{\overline{x_i}(1-\rho_1-\ldots-\rho_1)+R_i}{(1-\rho_1-\ldots-\rho_{i-1})(1-\rho_1-\ldots-\rho_i)} \quad i>1 \qquad (2.66)$$

2.7 . 3. M/G/1 Preemptive Resume Priority Queueing

For $i = 1$, we have

$$T_1 = \frac{\overline{x_1}(1-\rho_1)+R_1}{1-\rho_1} \qquad (2.67)$$