

Queueing Models

Simulation is often used in the analysis of queueing models. In a simple but typical queueing model, shown in Figure 6.1, customers arrive from time to time and join a *queue* (waiting line), are eventually served, and finally leave the system. The term “customer” refers to any type of entity that can be viewed as requesting “service” from a system. Therefore, many service facilities, production systems, repair and maintenance facilities, communications and computer systems, and transport and material-handling systems can be viewed as queueing systems.

Queueing models, whether solved mathematically or analyzed through simulation, provide the analyst with a powerful tool for designing and evaluating the performance of queueing systems. Typical measures of system performance include server utilization (percentage of time a server is busy), length of waiting lines, and delays of customers. Quite often, when designing or attempting to improve a queueing system, the analyst (or decision maker) is involved in tradeoffs between server utilization and customer satisfaction in terms of line lengths and delays. Queueing theory and simulation analysis are used to predict these measures of system performance as a function of the input parameters. The input parameters include the arrival rate of customers, the service demands of customers, the rate at which a server works, and the number and

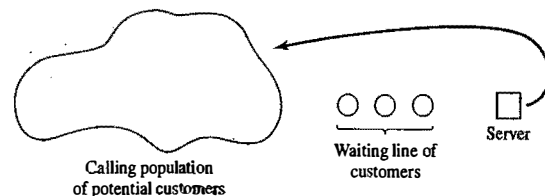


Figure 6.1 Simple queueing model.

arrangement of servers. To a certain degree, some of the input parameters are under management's direct control. Consequently, the performance measures could be under their indirect control, provided that the relationship between the performance measures and the input parameters is adequately understood for the given system.

For relatively simple systems, these performance measures can be computed mathematically—at great savings in time and expense as compared with the use of a simulation model—but, for realistic models of complex systems, simulation is usually required. Nevertheless, analytically tractable models, although usually requiring many simplifying assumptions, are valuable for rough-cut estimates of system performance. These rough-cut estimates may then be refined by use of a detailed and more realistic simulation model. Simple models are also useful for developing an understanding of the dynamic behavior of queueing systems and the relationships between various performance measures. This chapter will not develop the mathematical theory of queues but instead will discuss some of the well-known models. For an elementary treatment of queueing theory, the reader is referred to the survey chapters in Hillier and Lieberman [2005], Wagner [1975] or Winston [1997]. More extensive treatments with a view toward applications are given by Cooper [1990], Gross and Harris [1997], Hall [1991] and Nelson [1995]. The latter two texts especially emphasize engineering and management applications.

This chapter discusses the general characteristics of queues, the meanings and relationships of the important performance measures, estimation of the mean measures of performance from a simulation, the effect of varying the input parameters, and the mathematical solution of a small number of important and basic queueing models.

6.1 CHARACTERISTICS OF QUEUEING SYSTEMS

The key elements of a queueing system are the customers and servers. The term “customer” can refer to people, machines, trucks, mechanics, patients, pallets, airplanes, e-mail, cases, orders, or dirty clothes—anything that arrives at a facility and requires service. The term “server” might refer to receptionists, repairpersons, mechanics, tool-crib clerks, medical personnel, automatic storage and retrieval machines (e.g., cranes), runways at an airport, automatic packers, order pickers, CPUs in a computer, or washing machines—any resource (person, machine, etc.) that provides the requested service. Although the terminology employed will be that of a customer arriving to a service facility, sometimes the server moves to the customer; for example, a repairperson moving to a broken machine. This in no way invalidates the models but is merely a matter of terminology. Table 6.1 lists a number of different systems together with a subsystem consisting of “arriving customers” and one or more “servers.” The remainder of this section describes the elements of a queueing system in more detail.

6.1.1 The Calling Population

The population of potential customers, referred to as the *calling population*, may be assumed to be finite or infinite. For example, consider a bank of five machines that are curing tires. After an interval of time, a machine automatically opens and must be attended by a worker who removes the tire and puts an uncured tire into the machine. The machines are the “customers,” who “arrive” at the instant they automatically open. The worker is the “server,” who “serves” an open machine as soon as possible. The calling population is finite and consists of the five machines.

In systems with a large population of potential customers, the calling population is usually assumed to be infinite. For such systems, this assumption is usually innocuous and, furthermore, it might simplify the model. Examples of infinite populations include the potential customers of a restaurant, bank, or other similar service facility and also very large groups of machines serviced by a technician. Even though the actual

Table 6.1 Examples of Queueing Systems

System	Customers	Server(s)
Reception desk	People	Receptionist
Repair facility	Machines	Repairperson
Garage	Trucks	Mechanic
Tool crib	Mechanics	Tool-crib clerk
Hospital	Patients	Nurses
Warehouse	Pallets	Fork-lift Truck
Airport	Airplanes	Runway
Production line	Cases	Case-packer
Warehouse	Orders	Order-picker
Road network	Cars	Traffic light
Grocery	Shoppers	Checkout station
Laundry	Dirty linen	Washing machines/dryers
Job shop	Jobs	Machines/workers
Lumberyard	Trucks	Overhead crane
Sawmill	Logs	Saws
Computer	Jobs	CPU, disk, CDs
Telephone	Calls	Exchange
Ticket office	Football fans	Clerk
Mass transit	Riders	Buses, trains

population could be finite but large, it is generally safe to use infinite population models—provided that the number of customers being served or waiting for service at any given time is a small proportion of the population of potential customers.

The main difference between finite and infinite population models is how the arrival rate is defined. In an infinite population model, the arrival rate (i.e., the average number of arrivals per unit of time) is not affected by the number of customers who have left the calling population and joined the queueing system. When the arrival process is homogeneous over time (e.g., there are no “rush hours”), the arrival rate is usually assumed to be constant. On the other hand, for finite calling-population models, the arrival rate to the queueing system does depend on the number of customers being served and waiting. To take an extreme case, suppose that the calling population has one member, for example, a corporate jet. When the corporate jet is being serviced by the team of mechanics who are on duty 24 hours per day, the arrival rate is zero, because there are no other potential customers (jets) who can arrive at the service facility (team of mechanics). A more typical example is that of the five tire-curing machines serviced by a single worker. When all five are closed and curing a tire, the worker is idle and the arrival rate is at a maximum, but the instant a machine opens and requires service, the arrival rate decreases. At those times when all five are open (so four machines are waiting for service while the worker is attending the other one), the arrival rate is zero; that is, no arrival is possible until the worker finishes with a machine, in which case it returns to the calling population and becomes a potential arrival. It may seem odd that the arrival rate is at its maximum when all five machines are closed. But the arrival rate is defined as the expected number of arrivals in the next unit of time, so it becomes clear that this expectation is largest when all machines could potentially open in the next unit of time.

6.1.2 System Capacity

In many queueing systems, there is a limit to the number of customers that may be in the waiting line or system. For example, an automatic car wash might have room for only 10 cars to wait in line to enter the mechanism. It might be too dangerous (or illegal) for cars to wait in the street. An arriving customer who

finds the system full does not enter but returns immediately to the calling population. Some systems, such as concert ticket sales for students, may be considered as having unlimited capacity, since there are no limits on the number of students allowed to wait to purchase tickets. As will be seen later, when a system has limited capacity, a distinction is made between the arrival rate (i.e., the number of arrivals per time unit) and the effective arrival rate (i.e., the number who arrive and enter the system per time unit).

6.1.3 The Arrival Process

The arrival process for infinite-population models is usually characterized in terms of interarrival times of successive customers. Arrivals may occur at scheduled times or at random times. When at random times, the interarrival times are usually characterized by a probability distribution. In addition, customers may arrive one at a time or in batches. The batch may be of constant size or of random size.

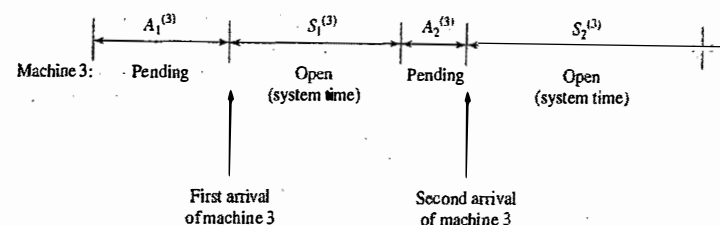
The most important model for random arrivals is the Poisson arrival process. If A_n represents the interarrival time between customer $n - 1$ and customer n (A_1 is the actual arrival time of the first customer), then, for a Poisson arrival process, A_n is exponentially distributed with mean $1/\lambda$ time units. The arrival rate is λ customers per time unit. The number of arrivals in a time interval of length t , say $N(t)$, has the Poisson distribution with mean λt customers. For further discussion of the relationship between the Poisson distribution and the exponential distribution, the reader is referred to Section 5.5.

The Poisson arrival process has been employed successfully as a model of the arrival of people to restaurants, drive-in banks, and other service facilities; the arrival of telephone calls to a call center; the arrival of demands, or orders for a service or product; and the arrival of failed components or machines to a repair facility.

A second important class of arrivals is scheduled arrivals, such as patients to a physician's office or scheduled airline flight arrivals to an airport. In this case, the interarrival times $\{A_n, n = 1, 2, \dots\}$ could be either *constant* or *constant plus or minus a small random amount* to represent early or late arrivals.

A third situation occurs when at least one customer is assumed to always be present in the queue, so that the server is never idle because of a lack of customers. For example, the “customers” may represent raw material for a product, and sufficient raw material is assumed to be always available.

For finite-population models, the arrival process is characterized in a completely different fashion. Define a customer as *pending* when that customer is outside the queueing system and a member of the potential calling population. For example, a tire-curing machine is “pending” when it is closed and curing a tire, and it becomes “not pending” the instant it opens and demands service from the worker. Define a *runtime* of a given customer as the length of time from departure from the queueing system until that customer's next arrival to the queue. Let $A_1^{(i)}, A_2^{(i)}, \dots$ be the successive runtimes of customer i , and let $S_1^{(i)}, S_2^{(i)}, \dots$ be the corresponding successive system times; that is, $S_n^{(i)}$ is the total time spent in system by customer i during the n th visit. Figure 6.2 illustrates these concepts for machine 3 in the tire-curing example. The total arrival process is the superposition of the arrival times of all customers. Figure 6.2 shows the first and second arrival of machine 3, but these two times are not necessarily two successive arrivals to the system. For instance,

**Figure 6.2** Arrival process for a finite-population model.

if it is assumed that all machines are pending at time 0, the first arrival to the system occurs at time $A_1 = \min\{A_1^{(1)}, A_1^{(2)}, A_1^{(3)}, A_1^{(4)}, A_1^{(5)}\}$. If $A_1 = A_1^{(2)}$, then machine 2 is the first arrival (i.e., the first to open) after time 0. As discussed earlier, the arrival rate is not constant but is a function of the number of pending customers.

One important application of finite-population models is the machine-repair problem. The machines are the customers, and a runtime is also called time to failure. When a machine fails, it "arrives" at the queueing system (the repair facility) and remains there until it is "served" (repaired). Times to failure for a given class of machine have been characterized by the exponential, the Weibull, and the gamma distributions. Models with an exponential runtime are sometimes analytically tractable; an example is given in Section 6.5. Successive times to failure are usually assumed to be statistically independent, but they could depend on other factors, such as the age of a machine since its last major overhaul.

6.1.4 Queue Behavior and Queue Discipline

Queue behavior refers to the actions of customers while in a queue waiting for service to begin. In some situations, there is a possibility that incoming customers will balk (leave when they see that the line is too long), renege (leave after being in the line when they see that the line is moving too slowly), or jockey (move from one line to another if they think they have chosen a slow line).

Queue discipline refers to the logical ordering of customers in a queue and determines which customer will be chosen for service when a server becomes free. Common queue disciplines include first-in-first-out (FIFO); last-in-first-out (LIFO); service in random order (SIRO); shortest processing time first (SPT); and service according to priority (PR). In a job shop, queue disciplines are sometimes based on due dates and on expected processing time for a given type of job. Notice that a FIFO queue discipline implies that services begin in the same order as arrivals, but that customers could leave the system in a different order because of different-length service times.

6.1.5 Service Times and the Service Mechanism

The service times of successive arrivals are denoted by $S_1, S_2, S_3, \dots$. They may be constant or of random duration. In the latter case, $\{S_1, S_2, S_3, \dots\}$ is usually characterized as a sequence of independent and identically distributed random variables. The exponential, Weibull, gamma, lognormal and truncated normal distributions have all been used successfully as models of service times in different situations. Sometimes services are identically distributed for all customers of a given type or class or priority, whereas customers of different types might have completely different service-time distributions. In addition, in some systems, service times depend upon the time of day or upon the length of the waiting line. For example, servers might work faster than usual when the waiting line is long, thus effectively reducing the service times.

A queueing system consists of a number of service centers and interconnecting queues. Each service center consists of some number of servers, c , working in parallel; that is, upon getting to the head of the line, a customer takes the first available server. Parallel service mechanisms are either single server ($c = 1$), multiple server ($1 < c < \infty$), or unlimited servers ($c = \infty$). A self-service facility is usually characterized as having an unlimited number of servers.

Example 6.1

Consider a discount warehouse where customers may either serve themselves or wait for one of three clerks, then finally leave after paying a single cashier. The system is represented by the flow diagram in Figure 6.3. The subsystem, consisting of queue 2 and service center 2, is shown in more detail in Figure 6.4. Other variations of service mechanisms include batch service (a server serving several customers simultaneously), and a customer's requiring several servers simultaneously. In the discount warehouse, a clerk might pick several small orders at the same time, but it may take two of the clerks to handle one heavy item.

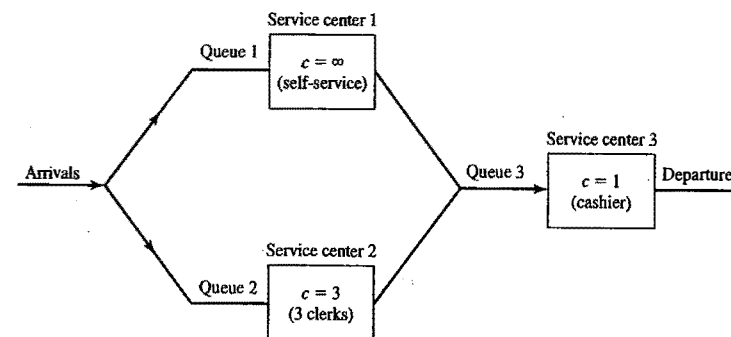


Figure 6.3 Discount warehouse with three service centers.

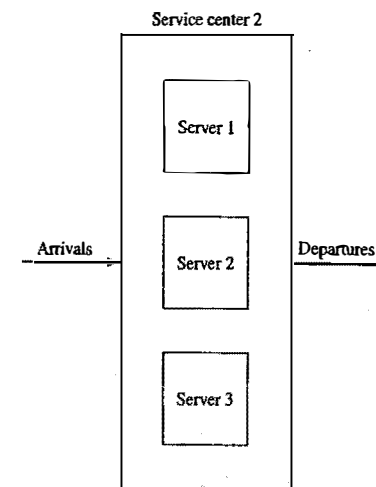


Figure 6.4 Service center 2, with $c = 3$ parallel servers.

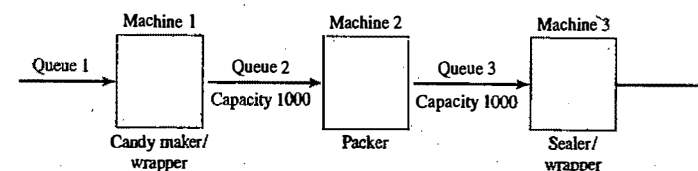


Figure 6.5 Candy-production line.

Example 6.2

A candy manufacturer has a production line that consists of three machines separated by inventory-in-process buffers. The first machine makes and wraps the individual pieces of candy, the second packs 50 pieces in a box, and the third machine seals and wraps the box. The two inventory buffers have capacities of 1000 boxes.

each. As illustrated by Figure 6.5, the system is modeled as having three service centers, each center having $c = 1$ server (a machine), with queue capacity constraints between machines. It is assumed that a sufficient supply of raw material is always available at the first queue. Because of the queue capacity constraints, machine 1 shuts down whenever its inventory buffer (queue 2) fills to capacity, and machine 2 shuts down whenever its buffer empties. In brief, the system consists of three single-server queues in series with queue capacity constraints and a continuous arrival stream at the first queue.

6.2 QUEUEING NOTATION

Recognizing the diversity of queueing systems, Kendall [1953] proposed a notational system for parallel server systems which has been widely adopted. An abridged version of this convention is based on the format $A/B/c/N/K$. These letters represent the following system characteristics:

A represents the interarrival-time distribution.

B represents the service-time distribution.

c represents the number of parallel servers.

N represents the system capacity.

K represents the size of the calling population.

Common symbols for A and B include M (exponential or Markov), D (constant or deterministic), E_k (Erlang of order k), PH (phase-type), H (hyperexponential), G (arbitrary or general), and GI (general independent).

For example, $M/M/1/\infty/\infty$ indicates a single-server system that has unlimited queue capacity and an infinite population of potential arrivals. The interarrival times and service times are exponentially distributed. When N and K are infinite, they may be dropped from the notation. For example, $M/M/1/\infty/\infty$ is often shortened to $M/M/1$. The tire-curing system can be initially represented by $G/G/1/5/5$.

Additional notation used throughout the remainder of this chapter for parallel server systems is listed in Table 6.2. The meanings may vary slightly from system to system. All systems will be assumed to have a FIFO queue discipline.

Table 6.2 Queueing Notation for Parallel Server Systems

P_n	Steady-state probability of having n customers in system
$P_n(t)$	Probability of n customers in system at time t
λ	Arrival rate
λ_e	Effective arrival rate
μ	Service rate of one server
ρ	Server utilization
A_n	Interarrival time between customers $n - 1$ and n
S_n	Service time of the n th arriving customer
W_n	Total time spent in system by the n th arriving customer
W_n^q	Total time spent in the waiting line by customer n
$L(t)$	The number of customers in system at time t
$L_q(t)$	The number of customers in queue at time t
L	Long-run time-average number of customers in system
L_q	Long-run time-average number of customers in queue
w	Long-run average time spent in system per customer
w_q	Long-run average time spent in queue per customer

6.3 LONG-RUN MEASURES OF PERFORMANCE OF QUEUEING SYSTEMS

The primary long-run measures of performance of queueing systems are the long-run time-average number of customers in the system (L) and in the queue (L_q), the long-run average time spent in system (w) and in the queue (w_q) per customer, and the server utilization, or proportion of time that a server is busy (ρ). The term "system" usually refers to the waiting line plus the service mechanism, but, in general, can refer to any subsystem of the queueing system; whereas the term "queue" refers to the waiting line alone. Other measures of performance of interest include the long-run proportion of customers who are delayed in queue longer than t_0 time units, the long-run proportion of customers turned away because of capacity constraints, and the long-run proportion of time the waiting line contains more than k_0 customers.

This section defines the major measures of performance for a general $G/G/c/N/K$ queueing system, discusses their relationships, and shows how they can be estimated from a simulation-run. There are two types of estimators: an ordinary sample average, and a time-integrated (or time-weighted) sample average.

6.3.1 Time-Average Number in System L

Consider a queueing system over a period of time T , and let $L(t)$ denote the number of customers in the system at time t . A simulation of such a system is shown in Figure 6.6.

Let T_i denote the total time during $[0, T]$ in which the system contained exactly i customers. In Figure 6.6, it is seen that $T_0 = 3$, $T_1 = 12$, $T_2 = 4$, and $T_3 = 1$. (The line segments whose lengths total $T_1 = 12$ are labelled " T_1 " in Figure 6.6, etc.) In general, $\sum_{i=0}^{\infty} T_i = T$. The time-weighted-average number in a system is defined by

$$\hat{L} = \frac{1}{T} \sum_{i=0}^{\infty} iT_i = \sum_{i=0}^{\infty} i \left(\frac{T_i}{T} \right) \quad (6.1)$$

For Figure 6.6, $\hat{L} = [0(3) + 1(12) + 2(4) + 3(1)]/20 = 23/20 = 1.15$ customers. Notice that T_i/T is the proportion of time the system contains exactly i customers. The estimator $\hat{L}$ is an example of a time-weighted average.

By considering Figure 6.6, it can be seen that the total area under the function $L(t)$ can be decomposed into rectangles of height i and length T_i . For example, the rectangle of area $3 \times T_3$ has base running from

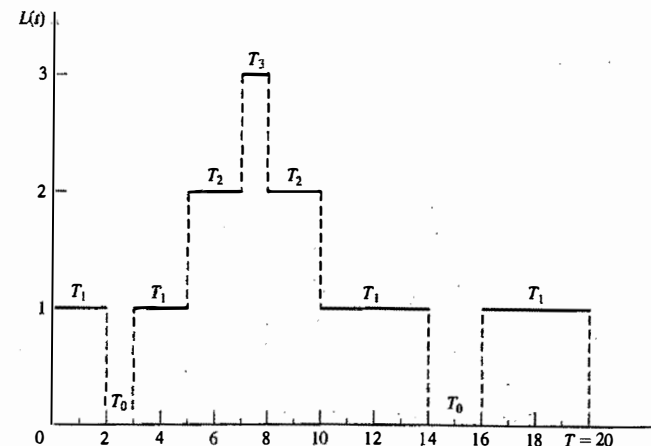


Figure 6.6 Number in system, $L(t)$, at time t .

$t = 7$ to $t = 8$ (thus $T_3 = 1$); however, most of the rectangles are broken into parts, such as the rectangle of area $2 \times T_2$ which has part of its base between $t = 5$ and $t = 7$ and the remainder from $t = 8$ to $t = 10$ (thus $T_2 = 2 + 2 = 4$). It follows that the total area is given by $\sum_{i=0}^{\infty} iT_i = \int_0^T L(t) dt$, and, therefore, that

$$\hat{L} = \frac{1}{T} \sum_{i=0}^{\infty} iT_i = \frac{1}{T} \int_0^T L(t) dt \quad (6.2)$$

The expressions in Equations (6.1) and (6.2) are always equal for any queueing system, regardless of the number of servers, the queue discipline, or any other special circumstances. Equation (6.2) justifies the terminology *time-integrated average*.

Many queueing systems exhibit a certain kind of long-run stability in terms of their average performance. For such systems, as time T gets large, the observed time-average number in the system $\hat{L}$ approaches a limiting value, say L , which is called the long-run time-average number in system—that is, with probability 1,

$$\hat{L} = \frac{1}{T} \int_0^T L(t) dt \rightarrow L \text{ as } T \rightarrow \infty \quad (6.3)$$

The estimator $\hat{L}$ is said to be strongly consistent for L . If simulation run length T is sufficiently long, the estimator $\hat{L}$ becomes arbitrarily close to L . Unfortunately, for $T < \infty$, $\hat{L}$ depends on the initial conditions at time 0.

Equations (6.2) and (6.3) can be applied to any subsystem of a queueing system as well as they can to the whole system. If $L_Q(t)$ denotes the number of customers waiting in line, and T_i^Q denotes the total time during $[0, T]$ in which exactly i customers are waiting in line, then

$$\hat{L}_Q = \frac{1}{T} \sum_{i=0}^{\infty} iT_i^Q = \frac{1}{T} \int_0^T L_Q(t) dt \rightarrow L_Q \text{ as } T \rightarrow \infty \quad (6.4)$$

where $\hat{L}_Q$ is the observed time-average number of customers waiting in line from time 0 to time T and L_Q is the long-run time-average number waiting in line.

Example 6.3

Suppose that Figure 6.6 represents a single-server queue—that is, a $G/G/1/N/K$ queueing system ($N \geq 3$, $K \geq 3$). Then the number of customers waiting in line is given by $L_Q(t)$ defined by

$$L_Q(t) = \begin{cases} 0 & \text{if } L(t) = 0 \\ L(t) - 1 & \text{if } L(t) \geq 1 \end{cases}$$

and shown in Figure 6.7. Thus, $T_0^Q = 5 + 10 = 15$, $T_1^Q = 2 + 2 = 4$, and $T_2^Q = 1$. Therefore,

$$\hat{L}_Q = \frac{0(15) + 1(4) + 2(1)}{20} = 0.3 \text{ customers}$$

6.3.2 Average Time Spent in System Per Customer w

If we simulate a queueing system for some period of time, say T , then we can record the time each customer spends in the system during $[0, T]$, say $W_1, W_2, \dots, W_N$, where N is the number of arrivals during $[0, T]$. The average

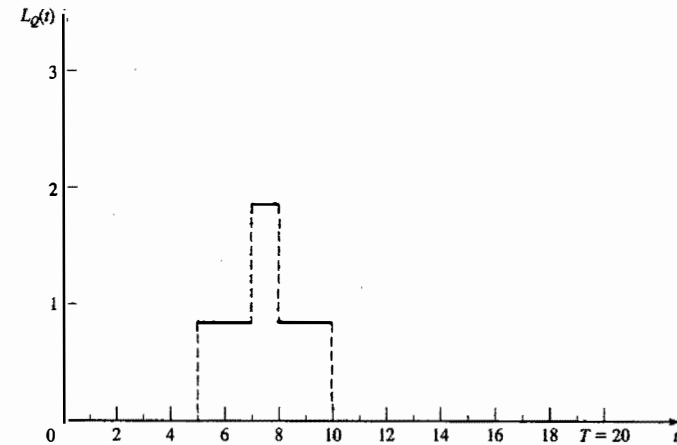


Figure 6.7 Number waiting in line, $L_Q(t)$, at time t .

time spent in system per customer, called the average system time, is given by the ordinary sample average

$$\hat{w} = \frac{1}{N} \sum_{i=1}^N W_i \quad (6.5)$$

For stable systems, as $N \rightarrow \infty$,

$$\hat{w} \rightarrow w \quad (6.6)$$

with probability 1, where w is called the long-run average system time.

If the system under consideration is the queue alone, Equations (6.5) and (6.6) are written as

$$\hat{w}_Q = \frac{1}{N} \sum_{i=1}^N W_i^Q \rightarrow w_Q \text{ as } N \rightarrow \infty \quad (6.7)$$

where W_i^Q is the total time customer i spends waiting in queue, $\hat{w}_Q$ is the observed average time spent in queue (called delay), and w_Q is the long-run average delay per customer. The estimators $\hat{w}$ and $\hat{w}_Q$ are influenced by initial conditions at time 0 and the run length T , analogously to $\hat{L}$.

Example 6.4

For the system history shown in Figure 6.6, $N = 5$ customers arrive, $W_1 = 2$, and $W_5 = 20 - 16 = 4$, but W_2, W_3 , and W_4 cannot be computed unless more is known about the system. Assume that the system has a single server and a FIFO queue discipline. This implies that customers will depart from the system in the same order in which they arrived. Each jump upward of $L(t)$ in Figure 6.6 represents an arrival. Arrivals occur at times 0, 3, 5, 7, and 16. Similarly, departures occur at times 2, 8, 10, and 14. (A departure may or may not have occurred at time 20.) Under these assumptions, it is apparent that $W_2 = 8 - 3 = 5$, $W_3 = 10 - 5 = 5$, $W_4 = 14 - 7 = 7$, and therefore

$$\hat{w} = \frac{2 + 5 + 5 + 7 + 4}{5} = \frac{23}{5} = 4.6 \text{ time units}$$

Thus, on the average, an arbitrary customer spends 4.6 time units in the system. As for time spent in the waiting line, it can be computed that $W_1^q = 0$, $W_2^q = 0$, $W_3^q = 8 - 5 = 3$, $W_4^q = 10 - 7 = 3$, and $W_5^q = 0$; thus,

$$\hat{w}_q = \frac{0+0+3+3+0}{5} = 1.2 \text{ time units}$$

6.3.3 The Conservation Equation: $L = \lambda w$

For the system exhibited in Figure 6.6, there were $N = 5$ arrivals in $T = 20$ time units, and thus the observed arrival rate was $\hat{\lambda} = N/T = 1/4$ customer per time unit. Recall that $\hat{L} = 1.15$ and $\hat{w} = 4.6$; hence, it follows that

$$\hat{L} = \hat{\lambda} \hat{w} \quad (6.8)$$

This relationship between L , λ , and w is not coincidental; it holds for almost all queueing systems or subsystems regardless of the number of servers, the queue discipline, or any other special circumstances. Allowing $T \rightarrow \infty$ and $N \rightarrow \infty$, Equation (6.8) becomes

$$L = \lambda w \quad (6.9)$$

where $\hat{\lambda} \rightarrow \lambda$, and λ is the long-run average arrival rate. Equation (6.9) is called a conservation equation and is usually attributed to Little [1961]. It says that the average number of customers in the system at an arbitrary point in time is equal to the average number of arrivals per time unit, times the average time spent in the system. For Figure 6.6, there is one arrival every 4 time units (on the average) and each arrival spends 4.6 time units in the system (on the average), so at an arbitrary point in time there will be $(1/4)(4.6) = 1.15$ customers present (on the average).

Equation (6.8) can also be derived by reconsidering Figure 6.6 in the following manner: Figure 6.8 shows system history, $L(t)$, exactly as in Figure 6.6, with each customer's time in the system, W_i , represented by a rectangle. This representation again assumes a single-server system with a FIFO queue discipline. The

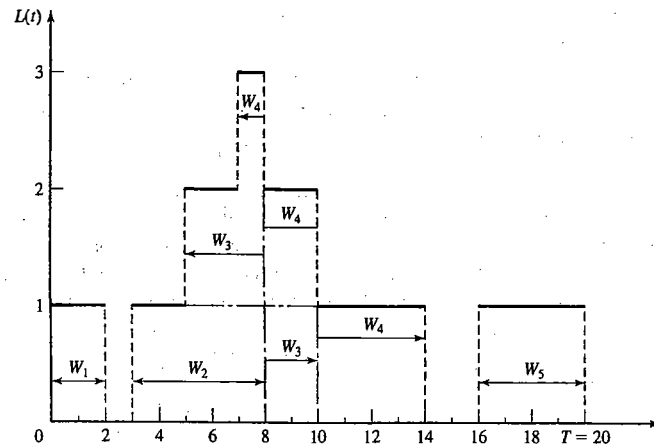


Figure 6.8 System times, W_i , for single-server FIFO system.

rectangles for the third and fourth customers are in two and three separate pieces, respectively. The i th rectangle has height 1 and length W_i for each $i = 1, 2, \dots, N$. It follows that the total system time of all customers is given by the total area under the number-in-system function, $L(t)$; that is,

$$\sum_{i=1}^N W_i = \int_0^T L(t) dt \quad (6.10)$$

Therefore, by combining Equations (6.2) and (6.5) with $\hat{\lambda} = N/T$, it follows that

$$\hat{L} = \frac{1}{T} \int_0^T L(t) dt = \frac{N}{T} \frac{1}{N} \sum_{i=1}^N W_i = \hat{\lambda} \hat{w}$$

which is Little's equation (6.8). The intuitive and informal derivation presented here depended on the single-server FIFO assumptions, but these assumptions are not necessary. In fact, Equation (6.10), which was the key to the derivation, holds (at least approximately) in great generality, and thus so do Equations (6.8) and (6.9). Exercises 14 and 15 ask the reader to derive Equations (6.10) and (6.8) under different assumptions.

Technical note: If, as defined in Section 6.3.2, W_i is the system time for customer i during $[0, T]$, then Equation (6.10) and hence Equation (6.8) hold exactly. Some authors choose to define W_i as total system time for customer i ; this change will affect the value of W_i only for those customers i who arrive before time T but do not depart until after time T (possibly customer 5 in Figure 6.8). With this change in definition, Equations (6.10) and (6.8) hold only approximately. Nevertheless, as $T \rightarrow \infty$ and $N \rightarrow \infty$, the error in Equation (6.8) decreases to zero, and, therefore, the conservation equation (6.9) for long-run measures of performance—namely, $L = \lambda w$ —holds exactly.

6.3.4 Server Utilization

Server utilization is defined as the proportion of time that a server is busy. Observed server utilization, denoted by $\hat{\rho}$, is defined over a specified time interval $[0, T]$. Long-run server utilization is denoted by ρ . For systems that exhibit long-run stability,

$$\hat{\rho} \rightarrow \rho \text{ as } T \rightarrow \infty$$

Example 6.5

Per Figure 6.6 or 6.8, and assuming that the system has a single server, it can be seen that the server utilization is $\hat{\rho} = (\text{total busy time})/T = (\sum_{i=1}^N T_i)/T = (T - T_0)/T = 17/20$.

Server utilization in G/G/1/ ∞ / ∞ queues

Consider any single-server queueing system with average arrival rate λ customers per time unit, average service time $E(S) = 1/\mu$ time units, and infinite queue capacity and calling population. Notice that $E(S) = 1/\mu$ implies that, when busy, the server is working at the rate μ customers per time unit, on the average; μ is called the service rate. The server alone is a subsystem that can be considered as a queueing system in itself; hence, the conservation Equation (6.9), $L = \lambda w$, can be applied to the server. For stable systems, the average arrival rate to the server, say λ_s , must be identical to the average arrival rate to the system, λ (certainly $\lambda_s \leq \lambda$ —customers cannot be served faster than they arrive—but, if $\lambda_s < \lambda$, then the waiting line would tend to grow in length at an average rate of

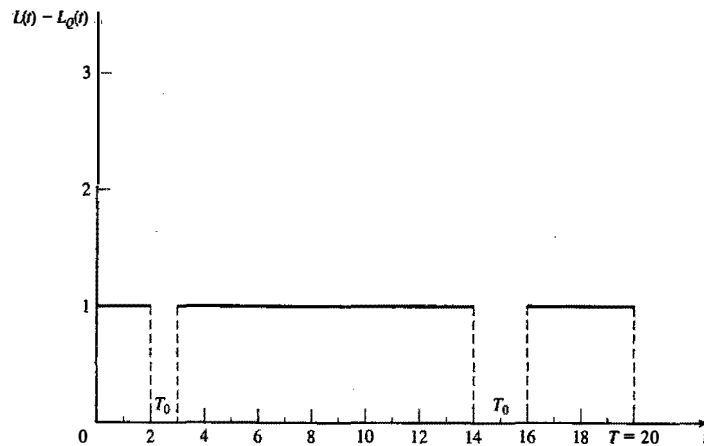


Figure 6.9 Number being served, $L(t) - L_Q(t)$, at time t .

$\lambda - \lambda_s$ customers per time unit, and so we would have an unstable system). For the server subsystem, the average system time is $w = E(S) = \mu^{-1}$. The actual number of customers in the server subsystem is either 0 or 1, as shown in Figure 6.9 for the system represented by Figure 6.6. Hence, the average number, $\hat{L}_s$, is given by

$$\hat{L}_s = \frac{1}{T} \int_0^T (L(t) - L_Q(t)) dt = \frac{T - T_0}{T}$$

In this case, $\hat{L}_s = 17/20 = \hat{\rho}$. In general, for a single-server queue, the average number of customers being served at an arbitrary point in time is equal to server utilization. As $T \rightarrow \infty$, $\hat{L}_s = \hat{\rho} \rightarrow L_s = \rho$. Combining these results into $L = \lambda w$ for the server subsystem yields

$$\rho = \lambda E(S) = \frac{\lambda}{\mu} \quad (6.11)$$

—that is, the long-run server utilization in a single-server queue is equal to the average arrival rate divided by the average service rate. For a single-server queue to be stable, the arrival rate λ must be less than the service rate μ :

$$\lambda < \mu$$

or

$$\rho = \frac{\lambda}{\mu} < 1 \quad (6.12)$$

If the arrival rate is greater than the service rate ($\lambda > \mu$), the server will eventually get further and further behind. After a time, the server will always be busy, and the waiting line will tend to grow in length at an average rate of $(\lambda - \mu)$ customers per time unit, because departures will be occurring at rate μ per time unit. For stable single-server systems ($\lambda < \mu$ or $\rho < 1$), long-run measures of performance such as average queue

length L_Q (and also L , w , and w_Q) are well defined and have meaning. For unstable systems ($\lambda > \mu$), long-run server utilization is 1, and long-run average queue length is infinite; that is,

$$\frac{1}{T} \int_0^T L_Q(t) dt \rightarrow +\infty \text{ as } T \rightarrow \infty$$

Similarly, $L = w = w_Q = \infty$. Therefore these long-run measures of performance are meaningless for unstable queues. The quantity λ/μ is also called the offered load and is a measure of the workload imposed on the system.

Server utilization in G/G/c/∞/∞ queues

Consider a queueing system with c identical servers in parallel. If an arriving customer finds more than one server idle, the customer chooses a server without favoring any particular server. (For example, the choice of server might be made at random.) Arrivals occur at rate λ from an infinite calling population, and each server works at rate μ customers per time unit. From Equation (6.9), $L = \lambda w$, applied to the server subsystem alone, an argument similar to the one given for a single server leads to the result that, for systems in statistical equilibrium, the average number of busy servers, say L_s , is given by

$$L_s = \lambda E(S) = \frac{\lambda}{\mu} \quad (6.13)$$

Clearly, $0 \leq L_s \leq c$. The long-run average server utilization is defined by

$$\rho = \frac{L_s}{c} = \frac{\lambda}{c\mu} \quad (6.14)$$

and so $0 \leq \rho \leq 1$. The utilization ρ can be interpreted as the proportion of time an arbitrary server is busy in the long run.

The maximum service rate of the G/G/c/∞/∞ system is $c\mu$, which occurs when all servers are busy. For the system to be stable, the average arrival rate λ must be less than the maximum service rate $c\mu$; that is, the system is stable if and only if

$$\lambda < c\mu \quad (6.15)$$

or, equivalently, if the offered load λ/μ is less than the number of servers c . If $\lambda > c\mu$, then arrivals are occurring, on the average, faster than the system can handle them, all servers will be continuously busy, and the waiting line will grow in length at an average rate of $(\lambda - c\mu)$ customers per time unit. Such a system is unstable, and the long-run performance measures (L , L_Q , w , and w_Q) are again meaningless for such systems.

Notice that Condition (6.15) generalizes Condition (6.12), and the equation for utilization for stable systems, Equation (6.14), generalizes Equation (6.11).

Equations (6.13) and (6.14) can also be applied when some servers work more than others, for example, when customers favor one server over others, or when certain servers serve customers only if all other servers are busy. In this case, the L_s given by Equation (6.13) is still the average number of busy servers, but ρ , as given by Equation (6.14), cannot be applied to an individual server. Instead, ρ must be interpreted as the average utilization of all servers.

Example 6.6

Customers arrive at random to a license bureau at a rate of $\lambda = 50$ customers per hour. Currently, there are 20 clerks, each serving $\mu = 5$ customers per hour on the average. Therefore the long-run, or steady-state, average utilization of a server, given by Equation (6.14), is

$$\rho = \frac{\lambda}{c\mu} = \frac{50}{20(5)} = 0.5$$

and the average number of busy servers is

$$L_s = \frac{\lambda}{\mu} = \frac{50}{5} = 10$$

Thus, in the long run, a typical clerk is busy serving customers only 50% of the time. The office manager asks whether the number of servers can be decreased. By Equation (6.15), it follows that, for the system to be stable, it is necessary for the number of servers to satisfy

$$c > \frac{\lambda}{\mu}$$

or $c > 50/5 = 10$. Thus, possibilities for the manager to consider include $c = 11$, or $c = 12$, or $c = 13$, Notice that $c \geq 11$ guarantees long-run stability only in the sense that all servers, when busy, can handle the incoming work load (i.e., $c\mu > \lambda$) on average. The office manager could well desire to have more than the minimum number of servers ($c = 11$) because of other factors, such as customer delays and length of the waiting line. A stable queue can still have very long lines on average.

Server utilization and system performance

As will be illustrated here and in later sections, system performance can vary widely for a given value of utilization, ρ . Consider a $G/G/1/\infty/\infty$ queue: that is, a single-server queue with arrival rate λ , service rate μ , and utilization $\rho = \lambda/\mu < 1$.

At one extreme, consider the $D/D/1$ queue, which has deterministic arrival and service times. Then all interarrival times $\{A_1, A_2, \dots\}$ are equal to $E(A) = 1/\lambda$, and all service times $\{S_1, S_2, \dots\}$ are equal to $E(S) = 1/\mu$. Assuming that a customer arrives to an empty system at time 0, the system evolves in a completely deterministic and predictable fashion, as shown in Figure 6.10. Observe that $L = \rho = \lambda/\mu$, $w = E(S) = \mu^{-1}$, and $L_Q = w_Q = 0$. By varying λ and μ , server utilization can assume any value between 0 and 1, yet there is never any line whatsoever. What, then, causes lines to build, if not a high server utilization? In general, it is the variability of interarrival and service times that causes lines to fluctuate in length.

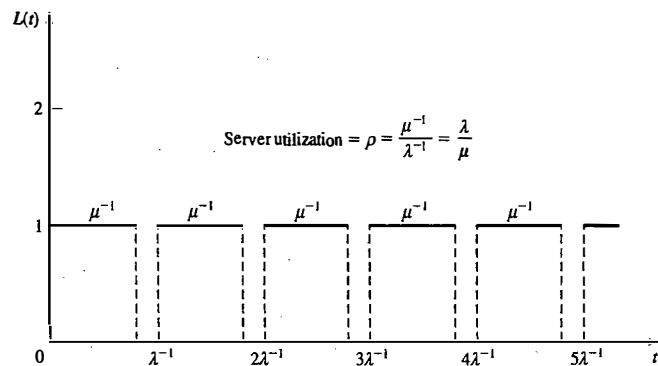


Figure 6.10 Deterministic queue (D/D/1).

Example 6.7

Consider a physician who schedules patients every 10 minutes and who spends S_i minutes with the i th patient, where

$$S_i = \begin{cases} 9 \text{ minutes with probability } 0.9 \\ 12 \text{ minutes with probability } 0.1 \end{cases}$$

Thus, arrivals are deterministic ($A_1 = A_2 = \dots = \lambda^{-1} = 10$) but services are stochastic (or probabilistic), with mean and variance given by

$$E(S_i) = 9(0.9) + 12(0.1) = 9.3 \text{ minutes}$$

and

$$\begin{aligned} V(S_i) &= E(S_i^2) - [E(S_i)]^2 \\ &= 9^2(0.9) + 12^2(0.1) - (9.3)^2 \\ &= 0.81 \text{ minutes}^2 \end{aligned}$$

Here, $\rho = \lambda/\mu = E(S)/E(A) = 9.3/10 = 0.93 < 1$, the system is stable, and the physician will be busy 93% of the time in the long run. In the short run, lines will not build up as long as patients require only 9 minutes of service, but, because of the variability in the service times, 10% of the patients will require 12 minutes, which in turn will cause a temporary line to form.

Suppose the system is simulated with service times, $S_1 = 9, S_2 = 12, S_3 = 9, S_4 = 9, S_5 = 9, \dots$. Assuming that at time 0 a patient arrived to find the doctor idle and subsequent patients arrived precisely at times 10, 20, 30, ..., the system evolves as in Figure 6.11. The delays in queue are $W_1^Q = W_2^Q = 0, W_3^Q = 22 - 20 = 2, W_4^Q = 31 - 30 = 1, W_5^Q = 0$. The occurrence of a relatively long service time (here $S_2 = 12$) caused a waiting line to form temporarily. In general, because of the variability of the interarrival and service distributions, relatively small interarrival times and relatively large service times occasionally do occur, and these in turn cause lines to lengthen. Conversely, the occurrence of a large interarrival time or a small service time will tend to shorten an existing waiting line. The relationship between utilization, service and interarrival variability, and system performance will be explored in more detail in Section 6.4.

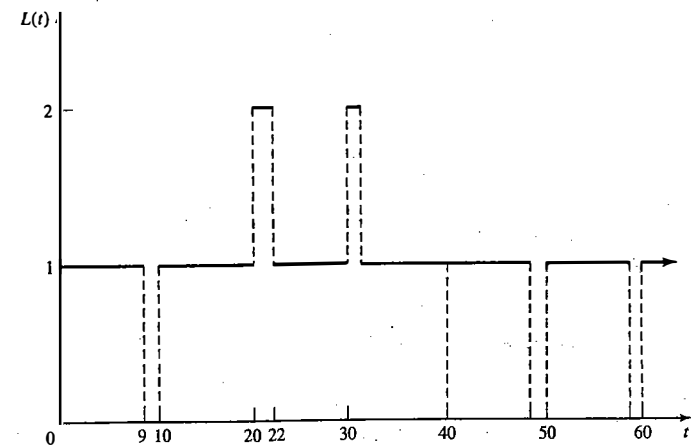


Figure 6.11 Number of patients in the doctor's office at time t .