NETWORK PERFORMANCE AND QUEUEING MODELS

Queueing models are an important class of mathematical models which can "predict" and explain certain aspects of the performance of networks and other systems where users statistically share resources. For communication networks, queueing models are often used to predict basic performance metrics such as blocking probability in circuit switched networks or packet delay in packet switched networks. Some of these models exactly predict the performance under some assumed traffic conditions, while others are only approximate. Some are statistical, some are deterministic. Some have simple analytical solutions, while others require numerical computations. In any case, since the actual traffic offered to a network is generally unknown or difficult to describe, the outcomes of these performance models only approximate reality to varying degrees of accuracy. Nevertheless, queueing analysis is an essential tool in the design, operation, and theory of networks.

Many of these models are applicable to a wide variety of network types, while others are quite specific. For instance, Markov chains are particularly useful in the evaluation of both circuit and packet switched networks. On the other hand, there are several modern models which are applicable to networks such as Asynchronous Transfer Mode (ATM) networks which provide virtual circuits with quality of service (QoS) guarantees.

We first define some of the most common performance metrics that queueing theory can predict. We will then describe how these predictions are used in network design and operation. After that, we will describe some of the most common models and important results.

Network Performance Metrics

There are a variety of QoS metrics, or measures, of a network's performance—for example, blocking probabilities, packet and message delay, delay jitter, throughput, and probability of loss. Roughly speaking, a blocking probability is the probability a new connection request is denied access to the network, packet (message) delay is a measure of how long the network takes to deliver the packet (message), jitter is a measure of how much variance there is between successive packet (message) deliveries, throughput is a measure of how much information is delivered per unit time, and loss probability is the probability a packet (message) is never delivered.

The relevance of a particular metric depends upon the type of network (e.g., connection-oriented or connectionless), the requirements of the applications which are using the network (e.g., real-time or non-real time), and goals of the network operator.

Blocking Probability. Blocking probability is a fundamental metric of most connection-oriented networks—that is, circuit-switched and virtual-circuit networks. In these networks, an application requests bandwidth in the form of a connection before transmitting data into the network. If insufficient resources are available for the connection (as determined by the type of network, a description of the desired resources, and network policy), the request is blocked.

The QoS of a single-rate circuit-switched network is often measured by the blocking probability, defined as the probability that a new call request is denied access to the network. Modern circuit-switched networks [e.g., integrated services digital network (ISDN)] provide multiple-rate circuits. The analysis of a multirate circuit network is more complex than that of a single-rate network, but the underlying queueing theory is quite similar. One key difference is that multirate networks are generally not evaluated based on a single blocking probability parameter, but rather on the set of blocking probabilities, one for each available rate.

Blocking probabilities are a function of the statistics of the traffic offered to the network (the call arrivals, the durations of calls, and the requested resources), the call admission control (CAC) policy which determines if a connection will be accepted, and the routing algorithm used to assign resources within the network. As such, blocking probabilities are often used to evaluate CAC and routing algorithms. Also, since user traffic can vary dramatically over time, blocking probabilities are often measured over time periods longer than a call duration but short enough so that traffic characteristics do not significantly change. Very often connection-oriented networks are evaluated based on their blocking probabilities during the busiest hour of the day.

Packet (Message) Delay and Loss. Classical data networks such as the Internet are typically used to transfer messages between computer applications. For these applications, the most basic metrics are message delay and loss.

Message delay is the total time the network takes to deliver the message from the time the first bit of the message enters the network to the time the last bit is delivered to the destination (if it is delivered). The message loss probability is the probability that a message offered to the network is never delivered.

Most networks do not transfer messages as their basic unit. The main reason for this is that simple queueing models show that message delays can be reduced by breaking down messages into smaller-sized units called *packets.* The Internet Protocol (IP) is the most common packet switching protocol and uses variable-sized packets with a minimum size of 20 bytes and a maximum size of 64,000 bytes. Packet networks can also use fixed-size packets. Continuous-time queueing models are used for variable-sized packet networks; discrete-time queueing models are used for fixed-sized packet networks. Most classical data networks use variable-sized packets. Modern fast packet networks use both variable-sized (Frame Relay) and fixed-size (ATM) packets. These latter networks can also transfer real-time applications such as voice and video, which tend to be very sensitive to delay and loss.

The most basic and fundamental measures of the performance of a packet network are packet delay and loss. Packet delay is the total time the network takes to deliver a packet from the time the first bit of the packet enters the network to the time it is delivered to the destination in its entirety. The packet loss probability is the probability that a packet offered to the network is never delivered.

Packet delay and loss are important for the obvious reasons that they strongly influence the total time needed to transfer a message as well as affecting the quality of real-time applications. For instance, delays in voice connections greater than a quarter of a second are quite perceptible and annoying to the participants. Note that message delays are functions of packet delays and packet losses. In case of packet loss, lost packets must be retransmitted, which delays the complete delivery of the message.

Delays and loss can occur for a variety of reasons but tend to dramatically increase as the amount of activity in the network increases; an excess of activity leads to congestion in all or part of the network, which causes queues to become backlogged or possibly run out of memory and overflow. Since usage is very often dynamic and hard to predict, delays and losses are time-varying random metrics. Given an appropriate statistical model for the offered traffic (packet arrivals and size), queueing models can predict the average packet delay, the variance of the packet delay, and a full statistical description of the delays and loss. These delays in turn can be used to predict message delays seen by the application. However, such an analysis can often be quite complex, and hence simplifying approximations are typically used. The situation becomes even more complex when packet networks are layered; that is, one packet network is used to send the packets of another packet network. A common example is the delivery of IP packets over an ATM or Frame Relay network.

Throughput. Throughput is a measure of the amount of data delivered per unit time.

Throughput is often measured in packets per second, but may also be measured in terms of bits per second. Note that throughput is a time-varying quantity and hence can be measured on different time scales.

Also note that if the packet loss rate is low, the throughput should be approximately equal to the rate at which bits are offered to the network. The maximum throughput is the maximum rate at which the packet loss rate and the packet delay are below predetermined acceptable levels.

Throughput is important for real-time applications. For example, good-quality video using MPEG-2 video compression requires a minimum throughput of 6 Mb/s.

Since throughput is strongly related to the amount of activity, a typical performance analysis will measure the packet delay and packet loss as functions of the throughput. Such an analysis is useful in a variety of situations. For instance, if we wish to compare two design options, we can say that one performs better if it has lower delay and loss for the same throughput.

Delay Jitter. An important metric in some virtual circuit packet networks is packet delay jitter. Jitter is a measure of the degree of variability in the time between successive packet deliveries in a virtual circuit. Excessive jitter can be highly detrimental to real-time applications such as video and voice. Packet delay and jitter may be traded off against each other. For example, if jitter is high, a large number of packets may have to be buffered at the destination in order to ensure a smooth play-out to the application.

ROLE OF QUEUEING ANALYSIS IN NETWORK DESIGN AND OPERATION

By modeling traffic, queueing models describe the system performance. These descriptions can then be used in the network planning, design, and operation.

Queueing models are widely used in network design. For instance, in the early planning stages of a new network, certain decisions have to be made. These decisions range from the most basic, such as deciding whether the network should be a packet network or a circuit network, to more complex decisions, such as the amount of bandwidth needed on the link, the topology of the network, and the protocols to be used (or invented).

There are generally innumerable choices for the physical layer of the network. For the purposes of this article, the physical layer may be thought of as a specified network topology which indicates which nodes, or switches, are connected to each other as well as a specification of the link bandwidths, or the rates at which nodes can communicate. Once the physical topology is determined, several other decisions must be made, including routing, flow control, and admission control policies. There are many options that can be considered in each of these decisions. As such, it is generally impossible for a human to reach an optimal answer (very often it is impossible for a computer to reach it since many network design problems fall into the category of NP-complete problems). Nevertheless, complex algorithms are used to design networks. Many of these algorithms use queueing models to evaluate the quality of a design and to decide on how to modify the current design to obtain a better design.

Queueing models are also needed in the operation of networks. Operational decisions based on queueing analysis occur on many different time scales. In the longest scale, the network operator or owner may decide to change some feature of the network—for example, purchase more bandwidth for a particular link. These decisions can be based on an analysis of the network and why the improved network should perform better—that is, generate more revenue, provide better service, and so on. On shorter time scales, the network operation can be modified in various ways. For instance, the routing decisions can be changed; the decisions as to which route to follow can be based on measured congestion in the network and some mapping of how congestion affects performance.

AN INTRODUCTION TO QUEUEING THEORY

Queueing theory is the mathematical framework used in the analysis and design of queueing systems. A queueing system is a system to which "customers" arrive in order to get "service." A bank branch in which tellers serve customer requests, a packet switch in a communication network which routes packets from its input ports to its output ports, and a statistical multiplexer which combines several traffic streams into one higher-rate stream are all examples of queueing systems. An important characteristic of these systems is the nondeterministic nature of the customer arrivals and their service demands. In the bank branch example above, it is not possible to determine the exact number and arrival times of customers, with certainty and a priori. Similarly the time required to serve a customer is typically unknown before the actual service takes place. Therefore, probabilistic models are employed to statistically characterize the arrival times and the service times of customers in a queueing system. Queueing theory is a branch of applied probability in which appropriate probability models are developed and utilized to predict system performance.

As in many engineering problems in which quantitative models are employed, queueing models involve tradeoffs between the capability of the model in reflecting the real system's properties and the model complexity. At one extreme are the simple queueing models that make simplifying assumptions about arrival and service statistics to ensure analytical tractability, and at the other extreme are complex and realistic computer models, sometimes developed from experimental observations of the real system, that require extensive development and simulation times. Often, a combination of analytical and computational techniques are used to evaluate the performance of the queueing system of interest. This article emphasizes analytical techniques because they have broader applicability and they provide valuable insights into the fundamental nature of queueing systems.

The simplest queueing system is a *single-server queue* which consists of a waiting room and a server as shown in Fig. 1. An arriving customer enters the waiting room and waits for its turn to receive service. In the context of a communication network, the customers are often either data packets or connection requests, and the waiting room is an electronic buffer (queue). The terms *packet, queue,* and *server* will be used to refer to the components of this queueing system. We will use the terms queue and buffer interchangeably.

An important issue in the design of a single-server queue is the *service discipline.* In first-in first-out (FIFO) service, the packets are served in the order of arrival with the earlier arrivals exiting the system before later arrivals. In last-in first-out (LIFO) service, the service order is reversed. The service discipline in a broadband network with multiple classes of traffic may be priority-based: Packets from a high-priority class are served before lower-priority packets.

Several system parameters must be specified to model a single-server queue. The *average arrival rate* λ (in packets per unit time) is a measure of the expected demand for the system. The *average service rate* μ is the average number of packets that are served per unit time by a busy server. The service rate determines the average speed of the server in units of packets per second—for example, the speed of transmission line in a multiplexer. The average time a packet spends in the server is given by $1/\mu$. Finally, the *buffer size* is the maximum number of packets that can be held in the buffer, including the packet receiving service.

The average arrival and service rates do not completely characterize the arrival and service processes. Probability distributions of these processes are required for system performance analysis. Several canonical distributions are commonly used for this purpose; these will be described in the following sections. First, an interesting and useful property that relates the average rates in an arbitrary queueing system to the average system occupancy and delay will be described.

LITTLE'S RESULT

Let λ denote the average arrival rate into a general queueing system (for systems in which arriving packets may be blocked from entering the system, λ must be replaced by the rate of packets entering the system). Suppose a packet P that arrives at this system in the steady state spends a time T in the system. [Steady state means that the system has been in operation for a sufficiently long time such that transient effects of an initially atypical (e.g., empty) system have subsided.] The value of T is random due to (a) the random number of packets that P finds in the system upon its arrival and (b) the random service requirements of these packets as well as that of P. Also suppose N is the number of packets in the system as seen by an independent observer at the steady state. J. Little has found that the average (expected) values of N and T are

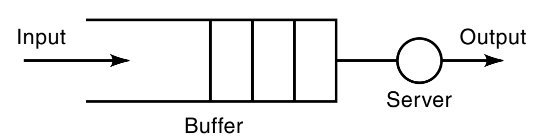

Figure 1. A single-server queueing system. Incoming customers wait in the buffer until they are processed by the server.

related as follows (1):

$$E(N) = \lambda E(T)$$

[We use the notation $E(X)$ to denote the expected value of a random variable X.] This relationship, known as *Little's result,* is perhaps the most useful result in queueing theory. The result is of surprising generality; it is valid irrespective of arrival or service distributions, the average service rate, the service discipline, and even the precise composition of the system. Little's result quantifies the intuition that congested systems [large $E(N)$] result in large delays and vice versa. The result also indicates that systems with large arrival rates tend to get more congested than those with lower arrival rates.

Little's result has found many applications in queueing theory. With appropriate definitions of a system and the quantities N, T, and λ, many interesting results can be obtained with economy. For instance, when the server of a single-server queue with average service rate μ is considered as the system of interest, one obtains

$$E(N) = \lambda/\mu = \rho$$

because the average time in the server is $1/\mu$. (The service rate must exceed the arrival rate for the system to be stable, a fact that will be elaborated upon when queueing delay is considered in detail.) Since the server can have 0 or 1 packets at a given time, $E(N)$ is the probability that $N = 1$. Thus ρ is the fraction of time the server is busy and is called the *server utilization.*

Little's result is particularly useful when either the average system occupancy or the average system delay is known and the other quantity is to be found. The reader is referred to Ref. 2 for an elementary and insightful proof of this result as well as many interesting applications to network performance analysis. Reference 3 provides a review of various generalizations of Little's result.

ARRIVAL AND SERVICE DISTRIBUTIONS IN QUEUEING

In order to obtain explicit performance results for queueing systems, one has to develop models for the statistics of arrival and service processes. Some of these models result in closed-form expressions, while others require numerical evaluation.

A natural means to model packet arrivals in queueing systems is through the use of counting processes. A counting process $N(t)$ is an integer-valued random process, whose value $N(t)$ is the number of events (packet arrivals) that occur up to (and including) time t. Thus $N(t) - N(s)$ is the number of arrivals during the time interval $(s, t]$. It is usually assumed that the process starts at time 0, so $N(0) = 0$.

A particular realization of the packet arrival process can be specified by the counting process $\{N(t): t \geq 0\}$ or, equivalently, by the sequence of packet arrival times $\{S_n: n = 1, 2, \ldots\}$, where S_n is the arrival time of the nth packet. The statement $N(t) = n$ is equivalent to the statement $S_n \leq t < S_{n+1}$. A third equivalent characterization of an arrival process is through the interarrival times $X_n = S_n - S_{n-1}$, where X_n is the time elapsed between the $(n - 1)$th and nth packet arrivals. This last characterization is the most common in queueing analysis. It is typically assumed that the interarrival times $X_1, X_2, \ldots$ are statistically independent and identically distributed (i.i.d.) random variables. That is, successive interarrivals are assumed to have no correlation. In this case the complete statistical description of the arrival process requires a single function to be specified, namely that characterizing the probabilistic behavior of the generic interarrival time X. Arrival processes with i.i.d. interarrival times are said to have the renewal property, because at each arrival instant the same probabilistic behavior is expected for the next arrival regardless of the past behavior of the process.

A distinction must be made between continuous-time and discrete-time queueing systems before specifying the interarrival statistics. In a continuous-time queueing system, arrivals and departures can occur at any time instant t. On the other hand, arrivals and departures are allowed to occur only at discrete time instants in a discrete-time queueing system. Discrete-time queueing systems will be considered in the section entitled “Discrete-Time Queues.” For continuous-time systems, interarrival statistics are often described in terms of a probability density function $f_X(x)$. This function quantifies the likelihood of the random variable X taking a value around x. In particular, for small $\delta > 0$, the probability $P(x \leq X < x + \delta)$ is approximately $\delta f_X(x)$. (This interpretation also justifies the term “density.”)

A common probability density function (pdf) in continuous-time queueing theory is the exponential density

$$f_X(x) = \lambda e^{-\lambda x}, \qquad x \geq 0$$

where λ is a positive parameter. X is said to be exponentially distributed if it has the pdf above. The expected value of X is $E(X) = 1/\lambda$. Thus, λ is the average number of arrivals per unit time and is called the (average) arrival rate. An arrival process with exponential interarrival pdf has the probability distribution

$$P(N(t) = k) = \frac{(\lambda t)^k}{k!} e^{-\lambda t}, \qquad k = 0, 1 \ldots$$

This is the *Poisson* distribution, and an arrival process with this distribution is called a Poisson process. Hence a queueing system has Poisson arrivals if (and only if) the interarrival times are exponentially distributed i.i.d. random variables.

The Poisson arrival process plays an important role in queueing theory because it simplifies the analysis of many queueing systems. While the traffic in real networks is almost certainly non-Poisson, networks designed using the Poisson traffic assumption have usually performed well. In a network with a large number of users each offering a small amount of traffic, the aggregate traffic tends to the Poisson distribution. In this sense, the role of the Poisson process in traffic modeling is analogous to that of the Gaussian process in noise modeling.

The exponential distribution is the only continuous distribution that has the following property. If X has exponential distribution, the conditional probability of the event $X \geq t + s$ given that $X \geq s$ is the unconditional probability of the event $X \geq t$; that is, $P(X \geq t + s \mid X \geq s) = P(X \geq t)$. If X is the waiting time until the occurrence of an event (say the arrival of a bus at a bus stop), according to this property, the amount of additional waiting is independent of the amount already spent waiting. This is known as the *memoryless property,* and it is the primary reason for the frequent use of Poisson traffic in queueing theory.

The second component of traffic characterization is the description of service times. Service times of packets in a queueing system may be random due to variable packet lengths (as in Internet and Ethernet). Even when the packet length is fixed (as in ATM), the amount of time a packet occupies the "head-of-line" in a queue may be random due to statistical sharing of transmission resources (e.g., in a switch). Therefore service times are commonly modeled as random variables in queueing analysis. The service times of packets are assumed to be statistically independent of the arrival times. In many systems the service times of successive packets may be accurately modeled as i.i.d. random variables. For these systems it suffices to specify a single probability density function $f_Y(y)$ for the service time of a generic packet. Exponential distribution $f_Y(y) = \mu e^{-\mu y}$, $y \geq 0$, is a frequent choice. Here $E(Y) = 1/\mu$ is the average service time, and μ is the average service rate of the server. Other service distributions are also common, including deterministic service for fixed-size packets served by a dedicated constant rate server.

BASIC QUEUEING MODELS

A queueing system is typically described using a shorthand notation of the form $A/B/L/K$. In this notation, A refers to the interarrival distribution, B refers to the service distribution, L denotes the number of servers in the system, and K denotes the size of the buffer. The last quantity is usually omitted when there is no limit on the number of customers that can be admitted to the system ($K = \infty$). Typical choices for the first two letters A and B are $\{M, D, G\}$, where M corresponds to exponentially distributed interarrivals or service (memoryless), D stands for a deterministic quantity, and G stands for a general (arbitrary) distribution. Examples of this notation are $M/M/1$, $M/D/1/K$, $G/M/K/K$, and so on. This notation provides a compact reference to the queueing system under consideration and is due to D. G. Kendall.

The simplest queueing system is the $M/M/1$, a single-server queue with Poisson arrivals, exponential service, and infinite buffer size. This system can be analyzed using continuous-time Markov chains, and many quantities of interest can be determined with ease. For example, the probability that there are n packets in this system in the steady state is given by

$$p_n = P(N = n) = \rho^n (1 - \rho), \qquad n = 0, 1, 2, \ldots$$

where $\rho = \lambda/\mu$ is the server utilization (see section entitled "Little's Result"). The average number of packets in the system (system occupancy) is then found to be

$$E(N) = \sum_{n=0}^{\infty} n p_n = \frac{\rho}{1 - \rho}$$

which is depicted in Fig. 2. Note that as the utilization increases, so does the system congestion, and sharply so beyond 80% utilization. This observation continues to hold for more general queueing systems and points out the need for excess service capacity to avoid system congestion and the associated delays. Little's result can be used to relate the system occupancy to average packet delay as

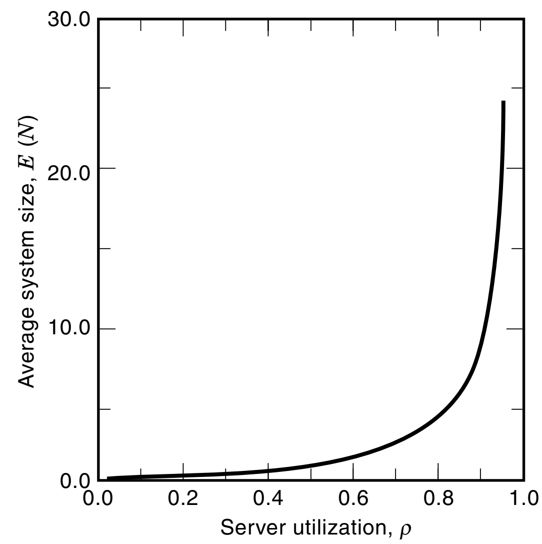

Figure 2. Average system occupancy of the $M/M/1$ queue as a function of the server utilization ρ. The system size increases slowly with ρ (server utilization) at first, then very sharply for $\rho \geq 0.8$.

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

which exhibits a similar behavior with increasing server utilization as the one shown in Fig. 2. It is observed that the average time a packet spends in the system is larger than the average service time $1/\mu$ by a factor $1/(1 - \rho)$ due to the waiting time in the buffer.

The average rate of packets processed by the queueing system is also known as the *throughput* of the system. Thus the server utilization ρ is also the normalized system throughput (throughput per average service time). The increase in system delay with increasing throughput is known as the *throughput-delay tradeoff.* While throughput is a measure of the revenue expected by the network operator, delay is a measure of the service quality the network customers get. A satisfactory resolution of this tradeoff is a critical task in network design.

As an application of the $M/M/1$ results above, consider a packet transmission system whose arrival rate is increased from λ to $b\lambda$ (where $b > 1$) while the service rate is increased from μ to $b\mu$. The server utilization remains the same; therefore the average number of packets in the system is not affected by the scale-up. However, the average packet delay is reduced by a factor b. That is, a transmission system b times as fast will accommodate b times as many packets per second at b times smaller delay. This is an important reason why queueing delays may not be as important in high-speed networks.

The example above also points out the benefit of statistical multiplexing in networks. Suppose there are b traffic streams each at rate λ packets per second and a total server capacity of $b\mu$ packets per second. In traditional time division multiplexing (TDM), each of the streams see an effective service rate of μ, while in statistical multiplexing the streams are merged into an aggregate stream of rate $b\lambda$ and a single server of rate $b\mu$ is employed. As a result, the packet delays are b times lower with statistical multiplexing. It is also seen that it is advantageous to merge waiting lines in a multiple server environment (such as a bank branch or a fast food enterprise). This observation continues to hold for arbitrary arrival and service statistics.

The analyses of the single-server, finite-buffer queueing system $M/M/1/s$ and the s-server queueing system $M/M/s$ utilize the same Markov chain formulation as the $M/M/1$. Specific results on delay and system occupancy can be found in standard texts on queueing theory (e.g., Ref. 2). An interesting variant of the $M/M/s$ system is the s-server loss system $M/M/s/s$. In this system there are s servers and no buffers. A packet that finds all the servers busy upon arrival does not enter the system and is lost. Hence the accepted packet rate into the system is lower than the arrival rate by a factor $1 - P_B$, where P_B is the packet loss (blocking) probability given by

$$P_B = \frac{(\lambda/\mu)^s/s!}{\sum_{i=0}^{s}(\lambda/\mu)^i/i!}$$

This equation is known as the Erlang-B formula and is very useful in dimensioning $M/M/s/s$ systems. The $M/M/s/s$ formulation finds a variety of applications in the design and analysis of telephone networks where it is used to estimate the call blocking probability as a function of traffic load per trunk λ/μ and the number of trunks s. It turns out that this loss formula is insensitive to service distribution and remains valid for $M/G/s/s$ systems with service rate μ (4).

The $M/G/1$ queueing system is a generalization of the $M/M/1$ system with an arbitrary probability density function $f_Y(y)$ for the service time. The first two moments of the service time $E(Y) = 1/\mu$ and $E(Y^2)$ are sufficient to obtain the average packet delay and system occupancy. The typical analysis involves an embedded Markov chain obtained by observing the system just after a service completion. At these time instants the memoryless property of interarrival times and the fresh start of a new packet service imply that the future evolution of the system state (the number of packets in the system) is independent of the past. An important expression in the analysis of $M/G/1$ queues is the Pollaczek–Khinchin formula, which relates the average system occupancy to the arrival and service parameters as

$$E(N) = \rho + \frac{\lambda^2 E(Y^2)}{2(1-\rho)}$$

which in conjunction with Little's result yields the average packet delay as

$$E(T) = \frac{1}{\mu} + \frac{\lambda E(Y^2)}{2(1-\rho)}$$

Note that these $M/G/1$ expressions reduce to the corresponding $M/M/1$ expressions since $E(Y^2) = 2/\mu^2$ for exponential service. It is also interesting to observe that deterministic service with $Y = 1/\mu$ minimizes both the average system occupancy and the average packet delay among all service distributions with the same service rate. For this $M/D/1$ queue, the second term in the delay expression above, which is the average waiting time in the buffer prior to service, is exactly 50% of the corresponding value for $M/M/1$.

In some multiple access networks the server is shared among many nodes. Token passing networks such as the token ring are examples of this type. Nodes in these networks can be modeled as $M/G/1$ queues with server vacations. In this model, the server takes a "vacation" after serving all the packets in a buffer. The amount of time the server spends in vacation is a random variable V with moments $E(V)$ and $E(V^2)$. The average system delay in this setting is given by the generalized Pollaczek–Khinchin formula

$$E(T) = \frac{1}{\mu} + \frac{\lambda E(Y^2)}{2(1-\rho)} + \frac{E(V^2)}{2E(V)}$$

Among all vacation queues with a given mean vacation period, the server with deterministic vacation causes the smallest delay and the smallest queue size.

PRIORITY QUEUEING

Modern broadband networks are designed to serve multiple classes of traffic, such as voice, video, data, and so on. Each such traffic class has a different delay requirement. Real-time traffic such as voice and video are sensitive to delay, while data traffic (e.g., e-mail and file transfer applications on Internet) is relatively delay-tolerant. When different traffic types share common network resources, such as transmission lines, routers, and so on, they may be given different service priorities to accommodate their service requirements. For example, in a single server system, delay-sensitive traffic may be served before delay-tolerant traffic. One possible scenario is to divide traffic into L priority classes with class i having priority over class $i + 1$ and to maintain a separate queue for each priority class. When a server becomes free, it starts serving a packet from the highest priority queue that is nonempty. In a nonpreemptive priority scheme, a packet service is completed without interruption even if a higher-priority packet arrives during that service. In preemptive priority schemes, packet service is interrupted with the arrival of a high-priority packet which starts receiving service immediately. The discussion below will focus on nonpreemptive priority, which is more appropriate for packet transmission by a single server.

The $M/M/1$ framework can be extended to multiple priorities as follows. For simplicity, we consider the case with two priorities. Let the Poisson arrival and exponential service rates of class i traffic be λ_i and μ_i, respectively, with $\rho_i = \lambda_i/\mu_i$. Assume for stability that $\rho_1 + \rho_2 < 1$. The average waiting time for a high-priority packet before it can start receiving service is

$$E(W_1) = E(R) + \frac{E(N_Q^1)}{\mu_1}$$

where R is the residual time of the packet being served at the time of arrival and N_Q^1 is the number of high-priority packets already in the queue. Little's result applied to the high-priority buffer yields $E(N_Q^1) = \lambda_1 E(W_1)$; therefore

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

For the low-priority class, two additional delay components are present. A low-priority packet has to wait for the service of all the packets that have arrived earlier, as well as those high-priority packets that arrive before this packet starts service. Then

$$E(W_2) = E(R) + \frac{E(N_Q^1)}{\mu_1} + \frac{E(N_Q^2)}{\mu_2} + \rho_1 E(W_2)$$

where the last term is due to tardy high-priority packets. Applying Little's result to both buffers one obtains

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

It is observed that low-priority packets experience a larger waiting time than high-priority packets by a factor $(1-\rho_1-\rho_2)^{-1}$. The final step to complete the delay analysis involves the calculation of the average residual time $E(R)$. The server is idle with probability $(1-\rho_1-\rho_2)$ and busy serving a class i packet with probability ρ_i. The memoryless property of the exponential service distribution then yields

$$E(R) = \frac{\rho_1}{\mu_1} + \frac{\rho_2}{\mu_2}$$

which can be used to obtain the waiting times explicitly. The average packet delays of the two classes are then found as

$$E(T_i) = E(W_i) + \frac{1}{\mu_i}, \qquad i = 1, 2$$

The average packet delay of an arbitrary packet is

$$E(T) = \frac{\lambda_1 E(T_1) + \lambda_2 E(T_2)}{\lambda_1 + \lambda_2}$$

This final delay expression can be used to verify that the average packet delay is minimized by assigning high priority to traffic with higher service rate. This is because a shorter packet will cause a lower waiting time for a longer packet than the case with reversed service orders. However, the packet delay averaged over service classes is often not the relevant performance measure due to difference in maximum tolerable delays for various traffic types. The priority assignments as well as the rate assignments must be chosen such that the delays for all traffic classes satisfy their requirements.

NETWORKS OF QUEUES

In a communication network, packets traverse a sequence of servers, such as transmission lines, switches, and store-and-forward nodes. Such a network may be modeled as an interconnection of queues where a packet departing from a server may enter another queue (or may depart from the network). An important characteristic of these systems is traffic mixing; different traffic streams interact with each other, making a compact traffic description very difficult.

There are two classes of queueing networks, open and closed networks, which will be treated separately below.

Open Queueing Networks

An open queueing network is a collection of queues with external arrivals and departures. Every packet entering an open network eventually departs from it.

In a queueing network, the basic assumption of statistical independence between interarrival times and packet service times that makes the analysis of a single queue possible no longer holds for the downstream queues. Consider, as an example, two single-server queues in tandem. The output packets from the first server join the queue for the second server, and packets leave the system once they are served by the second server. If the packet lengths (service times) are exponentially distributed and the external arrival process to the first queue is Poisson, the first queue can be analyzed by using the $M/M/1$ framework. An important result known as Burke's theorem, which applies not only to $M/M/1$ but more generally to $M/M/s$ and $M/M/\infty$ queues, implies that the departure process from the first queue is also Poisson. Therefore the second queue has Poisson arrivals. However, an interarrival time X at the second queue and the service time Y of the arriving packet at the first queue are statistically dependent (to see this, observe that $X \geq Y$). The packet length remains constant through the network; as a consequence, the service times at the second queue are not statistically independent of the interarrival times. Due to this correlation between arrivals and service, the second queue is not $M/M/1$ (or even $G/G/1$).

The exact analysis of the system with two queues in tandem is not known, because it is inherently difficult to account for the correlation illustrated above. To resolve this difficulty, an engineering approximation is employed in the analysis of queueing networks. This approximation is motivated by the fact that the input stream into a queue is typically a mixture of several packet streams. Kleinrock has suggested that this mixing effectively restores the independence of the arrival times and packet lengths. Consequently, Kleinrock's independence approximation adopts an $M/M/1$ model for each queue in a network. The approximation is accurate for networks with Poisson external traffic, exponentially distributed packet lengths, and a densely connected topology to ensure adequate mixing of traffic streams.

A typical application of Kleinrock's independence approximation in a queueing network is the delay analysis of a virtual circuit network. Here each packet stream l has a packet arrival rate λ_l and is assigned a path from the source node to the destination node in a given network topology. Let (i, j) denote the directed link from node i to node j and let $S(i, j)$ denote the set of streams that use this link. Each link is then modeled as an $M/M/1$ queue with the arrival rate

$$\lambda_{ij} = \sum_{l \in S(i,j)} \lambda_l$$

and service rate μ_{ij}. As a result, the average number of packets in the network at the steady state is given by

$$E(N) = \sum_{(i,j)} \frac{\rho_{ij}}{1-\rho_{ij}}$$

where $\rho_{ij} = \lambda_{ij}/\mu_{ij}$ is the utilization of link (i, j). Little's result, when applied to the whole network, yields the average packet delay as

$$E(T) = \frac{1}{\gamma} \sum_{(i,j)} \frac{\rho_{ij}}{1 - \rho_{ij}}$$

where $\gamma = \sum_l \lambda_l$ is the total arrival rate into the network. When processing and propagation delays are significant, the delay expression can be easily modified to take these effects into account. When the packet lengths are not exponentially distributed, the $M/G/1$ Pollaczek–Khinchin formula replaces the $M/M/1$ expressions above.

Jackson's theorem is a powerful result which shows that the delay expression above is exact, provided that packets are assigned anew independent and exponentially distributed service times in the queues they traverse. More generally, Jackson's theorem states that for a network with Poisson external arrivals the number of packets in each queue is statistically independent of those in all other queues. If $(n_1, n_2, \ldots, n_K)$ denotes the number of packets in a network of K queues, one has as the joint probability distribution

$$P(n_1, n_2, \ldots, n_K) = P_1(n_1)P_2(n_2) \cdots P_K(n_K)$$

where

$$P_j(n_j) = (1 - \rho_j)\rho_j^{n_j}, \qquad n_j = 0, 1, 2, \ldots$$

is the geometric distribution one would have for an $M/M/1$ queue in isolation, and ρ_j is the utilization of the jth server.

The importance of Jackson's theorem lies in the fact that it enables each queue in the network to be considered as an $M/M/1$ system in isolation, although the actual arrival process to the queue is, in general, non-Poisson. To see the latter, consider a single queue with external Poisson arrivals of rate λ_0 and a service rate $\mu \gg \lambda_0$. Suppose each packet completing service immediately returns to the queue with probability p and departs the system with probability $1 - p$. The total arrival rate into the queue is $\lambda_0/(1 - p)$, and from Jackson's theorem the number of packets in the system is geometrically distributed with parameter $\rho = \lambda_0/\mu(1 - p)$. Since each external arrival is likely to find the system empty, it induces another arrival after a short time with probability p (due to the short service time). Hence the aggregate arrival process is bursty and non-Poisson, although the system can be analyzed as if it were an $M/M/1$ system.

Closed Queueing Networks

A closed queueing network is a network in which a fixed number L of packets circulate without any external arrivals or departures. Such a model is usually employed to investigate the effect of limited system resources by implicitly assuming that each departure is immediately replaced by a new arrival. A common application of closed networks is in the analysis of window-based flow control schemes in packet-switched networks (5).

The typical quantity of interest in a closed network is the joint probability distribution of the number of packets in different queues. In a network of K queues these numbers $n_1, n_2, \ldots, n_K$ are clearly statistically dependent because their sum is a constant. Consequently, the isolation afforded by Jackson's theorem for open queueing networks does not hold for closed networks. However, the joint probability distribution of $(n_1, n_2, \ldots, n_K)$ can still be written as

$$P(n_1, n_2, \ldots, n_K) = c(L)P_1(n_1)P_2(n_2) \cdots P_K(n_K), \qquad n_1 + n_2 + \cdots + n_K = L$$

where $c(L) = (\sum_{n_1+\cdots+n_K=L} P_1(n_1)P_2(n_2) \cdots P_K(n_K))^{-1}$ and $P_i(n_i) = \rho_i^{n_i}$ with $\rho_i = \lambda_i/\mu_i$. Here μ_i is the service rate of the ith server and λ_i is the arrival rate to the ith queue. The arrival rates are determined by the routing probability matrix R whose (i, j)th entry R_{ij} is the probability that a packet leaving the ith queue joins the jth queue. The arrival rate vector $\Lambda = (\lambda_1, \lambda_2, \ldots, \lambda_K)^T$ is the solution to the equation $\Lambda = R^T \Lambda$. This equation has a unique nonzero solution within a multiplicative factor for a well-behaved R (the technical requirement is the irreducibility of the Markov chain with the transition probability matrix R). The multiplicative factor does not affect the probability distribution and can be arbitrarily chosen.

As an example, let us consider two queues in tandem with L circulating packets. Both servers have service rate μ. A packet served by the first server joins the second queue with probability r and returns back to the first queue with probability $1 - r$. The output packets from the second queue join the first queue. The arrival rates then satisfy $\lambda_2 = \lambda_1 r$. The packet occupancy distribution is then found as

$$P(n_1, L - n_1) = \frac{1 - r}{r^{-L} - r} r^{-n_1}, \qquad n_1 = 0, 1, \ldots, L$$

from which other quantities of interest, such as average queue occupancy and packet delay, can be determined.

DISCRETE-TIME QUEUES

Our discussion has so far focussed on continuous-time queues in which packet arrivals and service may occur at any time instant. There has been an increasing interest in broadband networks with fixed packet sizes over the last decade. This interest is primarily motivated by the asynchronous transfer mode (ATM) standard for broadband ISDN. ATM uses 53-byte packets (cells) and the network elements operate synchronously using time slots. The fixed packet size and time-slotted operation simplify the architecture and implementation of packet switches. Since the servers in such a network start service only at slot boundaries, the nature of queueing in a discrete-time queue is quite different from that in a continuous-time queue. In particular, the exact arrival times of packets are of secondary importance: The number of packets that arrive during a time slot is what affects the state of the system when observed at the beginning of the next time slot. For this reason, it is usually more convenient to describe the arrival process of a discrete-time queue in terms of the number of arrivals per slot instead of interarrival times. The packet service times are described in integer number of time slots. Hence the G-D-1 queue refers to a discrete-time queue with a general distribution on the number of arrivals per slot, a deterministic service time, and a single server. The G-G-1 queue is similarly defined.

The discrete-time G-D-1 queue is of primary interest in an ATM setting where each fixed-size packet needs a single time slot of service. The arrival process is i.i.d. from one slot to another and is specified by the probability distribution $P_A(n) = \Pr(n \text{ packet arrivals})$ or equivalently by the probability generating function $\phi_A(z) = \sum_{n=0}^{\infty} P_A(n)z^n$. The arrival rate is defined as the average number of packet arrivals per slot, $\lambda = \sum_n nP_A(n)$. For stability λ should not exceed unity. Let us observe the system at the beginning of each time slot. Let N_k be the number of packets in the system at the beginning of slot k, and let A_k be the number of arrivals that occur during that slot. The system occupancy is then described by the dynamic evolution equation

$$N_{k+1} = N_k - u(N_k) + A_k$$

where $u(n) = 1$ for $n > 0$ and $u(0) = 0$. The term $u(N_k)$ is the number of served packets in the kth slot. Since the number of arrivals A_k is independent of the state N_k, the sequence $\{N_k\}$ is a discrete-time Markov chain with transition probabilities

$$P_{ij} = \Pr(N_{k+1} = j \mid N_k = i) = P_A(j - i + u(i))$$

The steady-state distribution of this chain has the generating function

$$\phi_N(z) = (1-\lambda)\frac{(z-1)\phi_A(z)}{z - \phi_A(z)}$$

which can be inverse z-transformed, for a given arrival distribution, to obtain the steady-state system occupancy distribution. The average system occupancy can be found from $E(N) = \phi_N'(1)$ as

$$E(N) = \frac{\lambda}{2} + \frac{\sigma_A^2}{2(1-\lambda)}$$

where σ_A^2 is the variance of the arrival distribution. This is the discrete-time Pollaczek–Khinchin formula, and it shows that deterministic arrivals minimize average system occupancy and delay.

An important application of discrete-time queueing is the analysis of an input-queueing packet switch. This switch is an M-input M-output device with a queue per input port. Each incoming packet is assumed to be equally likely to be destined to any one of the M output ports. These packets have fixed size which equals the slot duration. The switch serves up to M head-of-line (HOL) packets every time slot, two HOL packets with the same destination cannot be served in the same time slot. The system has a first-in first-out (FIFO) service discipline for each input queue. This means that a HOL packet that cannot be served in a given slot makes it impossible for a subsequent packet in the same input queue to be served in that slot, even if the output request of the latter packet could be honored. This effect is known as *HOL blocking* and introduces a correlation between destinations of HOL packets. (Two HOL packets are more likely to have an output conflict than two non-HOL packets.) This correlation has to be taken into account in the performance analysis.

The input-queueing packet switch is a system of M discrete-time queues with correlated service. The performance of this switch has been analyzed in Refs. 6 and 7 for Bernoulli arrivals. In this arrival model each input port receives a new packet with probability λ in a time slot. Multiple packet arrivals at the same input port in the same time slot are not allowed. Consequently, the arrival rate is λ packets per port per slot. The analysis decomposes the switch into M independent queueing systems in which a HOL packet is served with probability q in each time slot. Therefore the service time of a HOL packet is geometrically distributed with parameter q. (Since the packet interarrivals are geometrically distributed as well, this queue is sometimes referred to as a Geom/Geom/1 queue.) The decomposition approximation is known to be accurate when the parameter q is calculated by taking the HOL effect into account. This calculation yields (6)

$$q = \frac{2(1-\lambda)}{2-\lambda}$$

The average number of packets per input port can then be found as

$$E(N) = \frac{\lambda(1-\lambda)}{q-\lambda}$$

and the average packet delay is obtained from Little's result as $E(T) = (1 - \lambda)/(q - \lambda)$.

The maximum throughput of this switch is defined as the traffic rate λ beyond which finite system size and packet delay cannot be supported. This can be calculated from $\lambda_{\max} = 2(1 - \lambda_{\max})/(2 - \lambda_{\max})$ as $\lambda_{\max} = 2 - \sqrt{2} \simeq 0.586$. HOL blocking and destination conflicts reduce the maximum switch throughput from 100% to 58.6%. If the correlation between HOL destinations can be eliminated (e.g., by dropping the HOL packets that cannot be immediately switched), the throughput can be improved to $1 - e^{-1} \simeq 63.2\%$ at the expense of packet loss (7). It has been shown recently that 100% throughput can be achieved if non-FIFO service disciplines are used (8).

FUTURE TRENDS IN QUEUEING ANALYSIS AND NETWORK PERFORMANCE

In this final section we outline some of the research issues in modern network engineering which are related to queueing analysis.

A nonprobabilistic characterization of arrival processes has been developed by Cruz (9,10). This model assumes that every arrival process obeys certain average rate and burstiness criteria. Namely for all time intervals $[s, t]$, the number of packets that enter the network during this time interval is upper bounded by $\sigma + \rho(t - s)$, where ρ is the long-term average packet rate and σ is a parameter that controls the size of allowed packet bursts. This deterministic traffic description allows a worst-case characterization of packet delay and system occupancy. This framework has been applied to flow control in broadband ISDN (11,12).

Another current issue in network traffic engineering is the characterization of correlation and burstiness in statistical traffic models. Data, voice, image, and video sources all exhibit a strong temporal correlation which is not well-modeled by traditional models. Several advanced models have been proposed to account for correlation, such as Markov modulated Poisson processes (MMPP), fluid models (13), spectral models (14), and so on. These models attempt to quantify traffic correlation and burstiness in a parsimonious manner that enables a performance analysis. A consensus is yet to emerge on the adequacy of these models for characterizing traffic in modern networks. For a discussion of these issues the reader is referred to Ref. 15.

A related research topic in network performance is measurement-based traffic modeling. Many real traffic traces, including measurements of Ethernet and Internet traffic, have been observed to exhibit strong and slowly decaying temporal correlation (16). Statistical analyses of measured traffic data in many different network settings suggest a self-similar nature to network traffic. That is, time-averaged traffic seems to exhibit a behavior that is independent of the time scale over a wide range of such time scales, from a few milliseconds to several hours. This behavior is quite different from that of traditional traffic models used in queueing analysis and requires further study. Network performance implications of self-similar traffic are largely unknown at present. While early results suggest dramatic differences with certain performance metrics (e.g., packet loss probability in finite buffers), a comprehensive understanding of the queueing behavior with self-similar traffic is yet to be developed.

Finally, the interaction between delay in communication networks as quantified by queueing theory and the fundamental limits to reliable information transfer rates as quantified by Shannon's information theory remains to be fully understood. There are some interesting early results in this context (17,18); however, a basic framework that unifies queueing and information theories for network performance analysis is quite far in the horizon.

BIBLIOGRAPHY

1. J. Little, A proof of the queueing $L = \lambda W$, *Oper. Res.,* **9** (3): 383–387, 1961.
2. D. P. Bertsekas and R. G. Gallager, *Data Networks,* 2nd ed., Englewood Cliffs, NJ: Prentice-Hall, 1992.
3. W. Whitt, A review of $L = \lambda W$ and extensions, *Queueing Syst.: Theory and Appl.,* **9**: 235–268, 1991.
4. S. M. Ross, *Stochastic Processes,* New York: Wiley, 1983.
5. M. Schwartz, *Telecommunication Networks: Protocols, Modeling, and Analysis,* Reading, MA: Addison-Wesley, 1987.
6. J. Y. Hui, *Switching and Traffic Theory for Integrated Broadband Networks,* Boston: Kluwer, 1990.
7. M. J. Karol, M. G. Hluchyj, and S. P. Morgan, Input versus output queueing on a space-division packet switch, *IEEE Trans. Commun.,* **35**: 1347–1356, 1987.
8. N. McKeown, V. Anantharam, and J. Walrand, Achieving 100% throughput in an input-queued switch, *Proc. IEEE Infocom '95,* **1**: 296–302, 1996.
9. R. L. Cruz, A calculus for network delay, part I: Network elements in isolation, *IEEE Trans. Inf. Theory,* **37**: 114–131, 1991.
10. R. L. Cruz, A calculus for network delay, part II: Network analysis, *IEEE Trans. Inf. Theory,* **37**: 132–141, 1991.
11. A. K. Parekh and R. G. Gallager, A generalized processor sharing approach to flow control in integrated services networks: The single-node case, *IEEE/ACM Trans. Netw.,* **1**: 344–357, 1993.
12. A. K. Parekh and R. G. Gallager, A generalized processor sharing approach to flow control in integrated services networks: The multiple node case, *IEEE/ACM Trans. Netw.,* **2**: 137–150, 1994.
13. D. Anick, D. Mitra, and M. M. Sondhi, Stochastic theory of data-handling systems with multiple sources, *Bell Syst. Tech. J.,* **61**: 1871–1893, 1982.
14. S. Q. Li and C. L. Hwang, Queue response to input correlation functions: discrete spectral analysis, *IEEE/ACM Trans. Netw.,* **1**: 522–533, 1993.
15. R. G. Gallager et al., Advances in the fundamentals of networking–part I: Bridging fundamental theory and networking, *IEEE J. Selected Areas Commun.,* **13**: 1995.
16. W. E. Leland et al., On the self-similar nature of Ethernet traffic (extended version), *IEEE/ACM Trans. Netw.,* **2**: 1–15, 1994.
17. I. E. Telatar and R. G. Gallager, Combining queueing theory with information theory for multiaccess, *IEEE J. Selected Areas Commun.,* **13**: 963–969, 1995.
18. V. Anantharam and S. Verdu, Bits through queues, *IEEE Trans. Inf. Theory,* **42**: 4–18, 1996.

MURAT AZIZOĞLU
University of Washington
RICHARD A. BARRY
MIT Lincoln Laboratory