Computer Systems Modelling
Queueing Theory

Simon Crosby*

Michaelmas Term 1995

1 Overview

What is a model?

A model in our context is a simplified description of a system which is used to gain insight into the behaviour of that system. We are particularly interested in quantifying the performance of systems, e.g. how fast, rather than in distinguishing correct and incorrect behaviour.

Why model?

We model systems which are complex or expensive so that we don't have to build them or change them in order to find out how they will perform. The more rapidly technology changes, the greater the advantage in understanding how component changes will affect overall system performance.

There are several instances in which models can be used:

1. A manufacturer may have a range of compatible systems with different performances. A model might be used to determine the best configuration for a particular application.
2. If an existing system is performing inadequately a model can be used to determine how to improve the system and give an indication of the performance of the resulting system.

*These notes also include material by R.J. Gibbens and I.M. Leslie

3. Fundamental design decisions may affect performance of a system. A model can be used as part of the design process to avoid bad decisions.

A model should contain only essential details. Determining what details are and aren't essential is often a tricky problem; it has been said that modelling is an art not a science. In general, the less details there are the easier a model is to manipulate; the more details there are the more confidence one has about the information the model yields. But too many parameters is bad if they are hard to measure or predict.

In any modelling exercise one must contend with two domains. These are the domain of system being modelled, in our case computer systems, and the domain of the model, often a mathematical domain. One must understand both domains in order to model a system. This seems difficult, but often a great deal of insight can be gained from very simple considerations.

In this course we will look at some techniques for modelling computer systems. In particular, we shall examine two approaches: *Queueing Theory* and *Operational Analysis*.

Queueing theory is mathematically rigorous and gives rise to exact (or sometimes approximate or asymptotic) solutions of mathematical models. It is based on the notion of stochastic processes which are processes in which the states are random variables. It thus draws upon probability theory. The treatment in this course will be quite shallow, a more detailed account can be found in Kleinrock *Queueing Systems — Volume I: Theory*, Wiley 1975.

Operational analysis, on the other hand, is a rather more down to earth approach in which one is only concerned with what is measurable or what can be derived directly from measurable quantities. The assumptions made by operational analysis tend to be weaker (and thus more general) than those made by queueing theory, but the solutions obtained are often inexact. The material in this course will be based on Lazowska, Zahorjan, Graham, and Sevcik, *Quantitative System Performance*, Prentice-Hall 1984.

There is a more recent (and inexpensive) book which covers all, or nearly all of the material in the course. It is probably the one worth buying. This is Leung's *Quantitative Analysis of Computer Systems*, Wiley 1988.

Before all this, however, we will talk in more general terms.

A toy problem

A company has a database running on a type A system. It runs like a drain. The manufacturer of the system can supply the company with a type B1 system immediately, and a type B2 system in the future. All of these systems are software compatible. Some measurements on the CPU and the single disk are made, and the resources consumed by a single transaction are tabulated below.

system type	CPU time (seconds)	disk time (seconds)	total (seconds)
A	4.6	4.0	8.6
B1	5.1	1.9	7.0
B2	3.1	1.9	5.0

The response time for a transaction under heavy load is currently well in excess of 8.6 seconds. It follows that transactions must also be experiencing delays while waiting for service. What should the company do?

The answer may or may not be obvious to you. In the current system (A), the CPU is the most heavily used device. The poor response time will be a result of people queueing at the CPU, that is, waiting to use the CPU. Speeding up the disk (as in system B1) won't help that a great deal. Slowing down the CPU will be disasterous. System B1 should be avoided; it is the worst of the three for this application. The company should wait until the B2 system is available. Performing "back of the envelope" analysis like this is quite simple providing one understands the basics of modelling techniques.

A comparison of techniques

We can distinquish at least four techniques for modelling systems:

1. *Simulation* – Simulation is considered by some purists to be "brute force" and not very insightful, but it is often much the simplest way of proceeding. It especially useful when discrete time or deterministic behaviour of a system is relevant. It is probably the most flexible, but also the most computationally expensive method. It can play an important part in verifying the accuracy of approximate analytical methods.

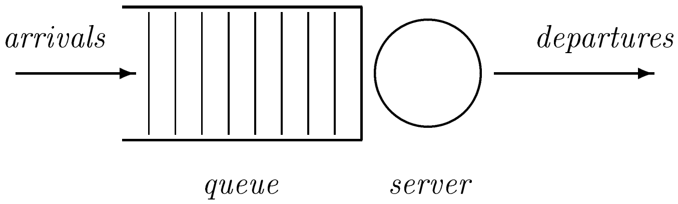
2. *Queueing theory* – This often yields closed form solutions to a number of problems. Limitations arise from the lack of flexibility; one must often make drastic assumptions about the system in question in order to make the problem tractable. Solutions are very insightful into system behaviour. It is difficult to model multiple component systems.

3. *Operational analysis* – Assumptions are weaker than queueing theory, but it is computationally more expensive (often must iterate to a solution). Multiple component systems are relatively easy to model.

4. *Back of the envelope* – Certainly the most efficient method. The insights gained depend upon the experience of the modeller; understanding of other methods is important.

2 Queueing Theory

The simplest system which queueing theory addresses is shown below. Customers arrive at a single *server*, wait in the queue if the server is busy, are then serviced and finally depart from the system. For example, the server might be a clerk in a bank or a transaction request in a database system.



Given a description of the arrival process, (which would include the average rate of arrival but would also include other information) and a description of the time that customers take to be served, the goal is to determine other properties of the system. Two such properties are a description of the queue length (including but not confined to the average) and a description of how busy the server is.

Queueing theory relies on the theory of stochastic processes, that is, processes whose states are described by random variables.

2.1 Stochastic Processes

We will begin by introducing some notation.

We write the probability that an event will occur as $P[event]$, that is, $P[X \leq .5]$ is the probability that a random variable X is less than or equal to 0.5. The probability that event A and event B will both occur is written as P[A,B]. The *conditional probability* of an event A occuring given that event B has occurred is written as $P[A \mid B]$ and is defined to be

$$P[A \mid B] \triangleq \frac{P[A, B]}{P[B]} .$$

In order to describe the *distribution* from which a random variable is selected, we have two functions, the *cumulative distribution function* (CDF) and the *probability density function* (pdf). The cumulative distribution function of a continuous random variable X is

$$F_X(x) \triangleq P[X \leq x] ,$$

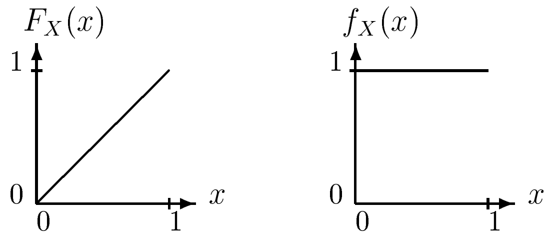
with $F_X(\infty) = 1$. The probability generating function is just the derivative of the CDF with respect to x:

$$f_X(x) \triangleq \frac{dF_X(x)}{dx}$$

with

$$F_X(x) = \int_{-\infty}^{x} f_X(y)dy\,, \quad \text{and} \quad \int_{-\infty}^{\infty} f_X(y)dy = 1.$$

Thus for a random variable X uniformly distributed from 0 to 1, the CDF and pdf would be



The *expected value* of X, $E[X]$ (sometimes called the average, or mean), is just

$$E[X] \triangleq \overline{X} \triangleq \int_{-\infty}^{\infty} x f_X(x)dx\,.$$

Note that $E[X+Y] = E[X] + E[Y]$. The expected value is also called the first *moment* of the distribution. The n^{th} *moment* is defined as

$$E[X^n] \triangleq \overline{X^n} \triangleq \int_{-\infty}^{\infty} x^n f_X(x)dx\,,$$

and the n^{th} *central moment* is

$$\overline{(X-\overline{X})^n} = \int_{-\infty}^{\infty} (x-\overline{X})^n f_X(x)dx\,.$$

The second central moment is known as the *variance*,

$$\sigma_X^2 \triangleq \overline{(X-\overline{X})^2} = \overline{X^2} - (\overline{X})^2.$$

The square root of the variance is σ_X and is called *standard deviation.* A dimensionless quantity, the *coefficient of variation*, is obtained by dividing the standard deviation by the average:

$$C_X \triangleq \frac{\sigma_X}{\overline{X}}\,.$$

Similar definitions apply to the case of discrete random variables taking values x_i ($i \in I$), say, such that $\sum_{i\in I} P[X = x_i] = 1$. The expected value of X is

$$E[X] = \sum_{i\in I} x_i P[X = x_i]$$

and similarly with the other moments the integration becomes a summation.

The notion of joint distributions is useful. The joint CDF of two random variables X and Y is

$$F_{XY}(x,y) \triangleq P[X \leq x, Y \leq y]\,,$$

and the joint pdf is given by

$$f_{XY}(x,y) \triangleq \frac{d^2 F_{XY}(x,y)}{dxdy}\,.$$

The random variables X and Y are said to be *independent* if the the joint pdf factors

$$f_{XY}(x,y) = f_X(x) f_Y(y)\,.$$

We can now describe a stochastic process. A *stochastic process* or *random function* is a function in time $X(t)$ whose values are random variables. A stochastic process might be described by a CDF which varies over time:

$$F_X(x,t) \triangleq P[X(t) \leq x]\,.$$

However, this may not tell the whole story. The random variable $X(t_1)$ may *not be independent* of the random variable $X(t_2)$, for two points in time t_1 and t_2; indeed if $X(t_1)$ and $X(t_2)$ were independent for all t_1 and t_2, $X(t)$ would not coincide with our intuitive notion of a process. We can give more information about a stochastic process by describing the joint distribution of the variable in different points in time. Consider a sequence of points in time $t_1, t_2, t_3 \ldots t_n$. Then

$$F_{X_1 X_2 X_3 \ldots X_n}(x_1, x_2, \ldots x_n; t_1, t_2, \ldots t_n) \triangleq P[X(t_1) \leq x_1, X(t_2) \leq x_2, \ldots X(t_n) \leq x_n]\,,$$

which we can write simply as $F_{\underline{\mathbf{X}}}(\underline{\mathbf{x}}; \underline{\mathbf{t}})$, where $\underline{\mathbf{x}} = (x_1, x_2, \ldots x_n)$ and $\underline{\mathbf{t}} = (t_1, t_2, \ldots, t_n)$. These joint distributions for every $\underline{\mathbf{x}}$, $\underline{\mathbf{t}}$ and n are in some sense the rules which govern the behaviour of the process.

There is an interesting class of processes whose joint distributions remain the same subject to a shift in time. The notation $\underline{\mathbf{t}} + \tau$ is used to indicate a shift of τ, and is equivalent to $(t_1 + \tau, t_2 + \tau, \ldots t_n + \tau)$. Given a process $X(t)$, if for every τ,

$$F_{\underline{\mathbf{X}}}(\underline{\mathbf{x}}; \underline{\mathbf{t}} + \tau) = F_{\underline{\mathbf{X}}}(\underline{\mathbf{x}}; \underline{\mathbf{t}})$$

then the process is said to be *stationary*, that is, the rules governing the process do not change in time.

As a simple example, consider the temperature in a room. We could describe a probability distribution for the temperature in the room at some point in time. But our notion of a process would be better served if we could describe the dependence of that temperature at that point in time on the temperature say ten minutes earlier. We could do this by way of a joint distribution. Now consider whether or not such a process is stationary;

that is does this dependence on the temperature ten minutes earlier in itself depend upon the absolute time. If the central heating always comes on at 9:00 am, then the expected temperature at 9:10 is likely to be greater than the temperature at 9:00. If we were considering the temperature at 3:00 pm and 3:10 pm, then we would say that they will be close to each other, that is they are definitely related, but we would not be able to say which was likely to be higher. The temperature of the room through time would thus *not* be a stationary process. Few, if any, of the systems we wish to model are stationary in this sense. However, we almost always assume they are. Thus we tacitly assume that any changes in the governing parameters of the system occur on a timescale much slower than that of the individual events.

The information required to specify the behaviour of even a stationary stochastic function is potentially unbounded. Quite simply, the value that a random variable takes on at any given time may be influenced by the values that it had over an unbounded sequence of previous points in time. Processes such as these are hardly tractable. We need to define a class (or classes) of processes with more restrictions.

2.2 Markov Chains

A stochastic process can have a discrete or continuous state space, and it can run in discrete or continuous time. The temperature in a room is an example of a continuous state space; the number of customers in a queue is an example of a discrete state space. If a process runs over discrete time, it can be represented as a sequence of random variables, $X_1, X_2, \ldots$; in continuous time we use the more general notation $X(t)$.

Markov chains have discrete state spaces, and can run in either continuous or discrete time. We consider discrete time first.

A sequence of random variables $X_1, X_2, \ldots$ forms a *Markov chain* if the probability that the next state is x_{n+1} depends only upon the current state x_n and not upon any previous state. That is, we have a random sequence in which the dependency extends backwards one unit in time. The entire way in which the past history affects the future of the process is summarized in one state.

In continuous time we allow transitions between states to occur at any point in time. However, the way in which history can affect the future of the process is again only described by the state we are in and *not by how long we have been there.* (Consider a telephone call. Does the time that a person has been on the phone affect the probability that he or she will hang up in the next second?) This property of not knowing how long we have been in a state (or rather not allowing it to affect the process) is referred to as the *memoryless* property.

The restrictions imposed on Markov chains make them easy to analyse but difficult to apply without making assumptions.

Before examining some particular types of Markov chain, we will look in more detail at the memoryless property. Consider the case of continuous time. We have said that when we change state depends only upon which state we are in, not on how long we have been there. Let τ be a random variable describing the time spent in a particular state. If we have been in that state for a time s, then what is the probability that we will remain in that state for a time $s+t$? We know it cannot depend upon s so

$$P[\tau > s+t \mid \tau > s] = h(t)\,.$$

For some function $h(t)$ not depending on s. From the definition of conditional probability,

$$\begin{aligned} P[\tau > s+t \mid \tau > s] &= \frac{P[\tau > s+t, \tau > s]}{P[\tau > s]} \\ &= \frac{P[\tau > s+t]}{P[\tau > s]} \\ &= h(t)\,. \end{aligned}$$

Therefore

$$P[\tau > s+t] = P[\tau > s]\, h(t)\,.$$

Now set $s = 0$, and since $P[\tau > 0] = 1$, we have

$$P[\tau > t] = h(t)$$

so that

$$P[\tau > s+t] = P[\tau > s]P[\tau > t]\,. \tag{1}$$

Now the pdf of τ is given by

$$\begin{aligned} f_\tau(t) &= \frac{d}{dt}P[\tau \le t] \\ &= \frac{d}{dt}\left(1 - P[\tau > t]\right) \\ &= -\frac{d}{dt}P[\tau > t]\,. \end{aligned} \tag{2}$$

Differentiating (1) with respect to s gives,

$$\begin{aligned} \frac{d}{ds}P[\tau > s+t] &= \frac{d}{ds}P[\tau > s]P[\tau > t] \\ &= -f_\tau(s)P[\tau > t]\,. \end{aligned}$$

Set $s = 0$ and divide by $P[\tau > t]$ to give,

$$\frac{dP[\tau > t]}{P[\tau > t]} = -f_\tau(0)ds\,.$$

Now integrate from 0 to t,

$$\begin{aligned} log_e P[\tau > t] &= -f_\tau(0)t \\ P[\tau > t] &= e^{-f_\tau(0)t}\,. \end{aligned}$$

From (2) above,

$$f_\tau(t) = f_\tau(0)e^{-f_\tau(0)t}$$

It is customary to use λ for $f_\tau(0)$, giving the familiar exponential distribution

$$f_\tau(t) = \lambda e^{-\lambda t}\,.$$

The mean of this distribution, that is the time we would expect to occupy a particular state is

$$\begin{aligned} E[\tau] &= \int_0^\infty t\lambda e^{-\lambda t}dt \\ &= -\lambda\left(\frac{\partial}{\partial\lambda}\int_0^\infty e^{-\lambda t}dt\right) \\ &= -\lambda\frac{\partial}{\partial\lambda}\left(\frac{1}{\lambda}\right) \\ &= -\lambda\left(-\frac{1}{\lambda^2}\right) \\ &= \frac{1}{\lambda} \end{aligned}$$

and with similar dazzling integration techniques,

$$\begin{aligned} E[\tau^2] &= \int_0^\infty t^2\lambda e^{-\lambda t}dt \\ &= \frac{2}{\lambda^2} \end{aligned}$$

so that the coefficient of variation is

$$\begin{aligned} C_\tau &\triangleq \frac{\sigma_\tau}{\overline{\tau}} \\ &= \frac{\sqrt{\overline{\tau^2} - (\overline{\tau})^2}}{\overline{\tau}} \\ &= \frac{\sqrt{\frac{2}{\lambda^2} - (\frac{1}{\lambda})^2}}{\frac{1}{\lambda}} \\ &= 1\,. \end{aligned}$$

The important fact from all this is that a memoryless distribution is necessarily the exponential distribution. Choosing an exponential distribution in a model is somewhat limiting;

there is only one parameter to choose and the coefficient of variation is always unity. However, there is a certain simplicity; the parameter λ can be interpreted directly as a rate.

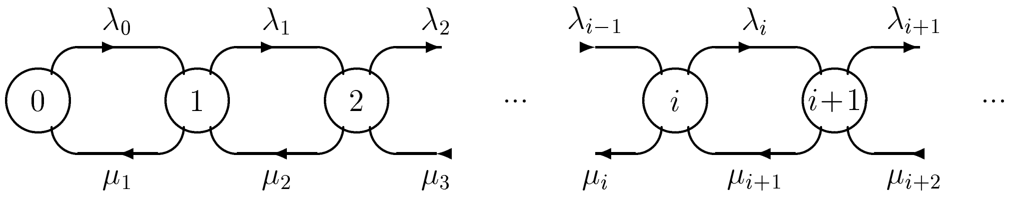
This notion of rate is useful in the following way. Suppose we have a state with two possible exit transitions to other states, and that the events associated with these transitions are e_1 and e_2. That is, we will exit from the state if either e_1 or e_2 occurs. If the pdfs describing the time at which these events occur are both exponential distributions with rates λ_1 and λ_2, then the pdf describing the time at which we will leave the state is also exponential with rate $\lambda_1 + \lambda_2$. This is perhaps not an earth shattering result, but it cannot be applied to more general distributions.

2.3 Birth Death Processes

There is a type of Markov chain which is particularly simple, but useful enough to model the single server queue above (as well as some more complex systems). The state space in a birth death process is the set of non-negative integers which correspond to the population. Changes are only allowed between adjacent states, i.e. there can only be a single birth or death at any transition. So that, if $X_n = i$ is the state at time n then

$$X_{n+1} = \begin{cases} i+1 & \text{birth (or arrival)} \\ i-1 & \text{death (or departure)} \end{cases}$$

We can thus model a birth death process as a chain of states with transitions linking them. The birth and death rate can be a function of the state we are in, but we must use an exponential distribution for state residency. We will use λ_i to represent the birth rate in state i, and μ_i to represent the death rate in state i.



The *state transition diagram* for a birth death process is shown above. Each circle represents a state, labelled with the value of the population to which the state corresponds. The arcs represent transitions between states and are labelled with their transition rate.

Solving this model is done by determining the state probabilities, that is, by determining what the probability is of being in each particular state. We denote by $P_i(t)$ the probability of being in state i at time t. These state probabilities will be a function of initial conditions and time, but we are interested in the long term probabilities after the system has reached an *equilibrium*. These probabilities are independent of the initial conditions.

More precisely, a system reaches equilibrium if, for all i,

$$\lim_{t\to\infty} P_i(t) = p_i \qquad \text{exisits}\,.$$

Here we will not be concerned with proving whether systems reach equilibrium, although we will indicate conditions where systems definitely will not reach an equilibrium. So, let us consider the birth death system shown above. If the system reaches equilibrium then each state has associated with it a probability. Furthermore we can deduce that transitions rates in and out of any state must be equal.

Consider the transitions into state i. These are births in state $i-1$ and deaths in state $i+1$. Thus we will enter state i at a rate $p_{i-1}\lambda_{i-1} + p_{i+1}\mu_{i+1}$. Similarly the rate of exit from the state is $p_i\lambda_i + p_i\mu_i$. Therefore we have a set of equations

$$p_{i-1}\lambda_{i-1} + p_{i+1}\mu_{i+1} = p_i\lambda_i + p_i\mu_i \;\; \text{for } i > 0\,.$$

We can also consider the transitions across a line between states i and $i+1$. These must also balance, since the number of transitions from the higher states, i.e. states representing a population of $i+1$ or more to the lower states must equal the number of transitions from the lower to the higher. Thus,

$$p_i\lambda_i = p_{i+1}\mu_{i+1} \;\; \text{for } i \geq 0$$

which are actually an equivalent set of equations to the above but are easier to deal with:

$$p_{i+1} = \frac{\lambda_i}{\mu_{i+1}} p_i$$

which gives rise to the product solution

$$p_k = p_0 \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}} \;\; \text{for } k \geq 0$$

and since the sum of state probabilities must be unity,

$$\begin{aligned} p_0 + \sum_{k=1}^{\infty} p_k &= 1 \\ p_0 + \sum_{k=1}^{\infty} p_0 \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}} &= 1 \\ p_o \left[1 + \sum_{k=1}^{\infty} \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}} \right] &= 1 \end{aligned}$$

so that

$$p_0 = \left[1 + \sum_{k=1}^{\infty} \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}}\right]^{-1}$$

$$p_k = p_0 \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}}$$

which are known as the *general flow balance equations.* It may be shown that an equilibrium distribution p_k always exists if $\frac{\lambda_k}{\mu_k} < 1$ for all $k > K$ for some fixed K.

We are now in a position to examine some simple queueing systems. Kleinrock uses a classification system (due to Kendall) which describes a queueing system by 5 parameters. These are

1. the arrival process (interarrival time distribution)
2. the service time distribution
3. the number of servers
4. the maximum allowable queue size
5. the total customer population

If the latter two items are not specified they are assumed to be infinite. The arrival process and service time distributions are described by a single letter. For example M is used to indicate an exponential (or Memoryless or Markov) distribution while G is used to indicate a general distribution. Thus a M/M/1/∞/∞ queueing system has one server, no limit on the number of customers in the queue, an infinite customer population, exponentially distributed service times and exponentially distributed interarrival times. An arrival process with exponentially distributed interarrival times is called a *Poisson* process. Here the Poisson *distribution* is concerned with the number of arrivals in a given time, not with the time between arrivals. Thus although the two distributions can be used to describe the same process, they are not the same. Since we can leave off the last two items when they are infinite, this system is more usually known as the M/M/1 queueing system. It is the simplest queueing system.

2.4 The M/M/1 Queue

We can model an M/M/1 queueing system as a birth death process since we are dealing with exponential distributions. Births represent arrivals into the system, deaths represent departures from the server. We have a single server, so the service rate is not a function of queue size. Thus for all $k > 1, \mu_k = \mu$. (By contrast, in a multiple server system the service rate is at a maximum when all the servers are busy, that is when the number of customers in the system is greater than or equal to the number of servers.) The population is infinite so that the fact there are customers in the system has no effect on the arrival rate. Thus for all $k, \lambda_k = \lambda$. (By contrast, in finite population systems, if every possible customer is in the system, no further customers can arrive until one departs.)

From the flow balance equations,

$$\begin{aligned} p_k &= p_0 \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}} \\ &= p_0 \left(\frac{\lambda}{\mu}\right)^k . \end{aligned}$$

where, writing $\rho = \frac{\lambda}{\mu}$,

$$\begin{aligned} p_0 &= \frac{1}{1+\sum_{k=1}^{\infty} \rho^k} \\ &= \frac{1}{1+\rho\sum_{k=0}^{\infty} \rho^k} \\ &= \frac{1}{1+\rho\left(\frac{1}{1-\rho}\right)} \\ &= 1-\rho \end{aligned}$$

Now, p_0 is the probability that there is no customer in the system and $1 - p_0 = \rho$ is the *utilization* of the server. Also, we have a geometric distribution for the number in the system

$$p_k = (1-\rho)\rho^k, \qquad k = 0, 1, 2, \ldots$$

If $\rho > 1$, i.e. if $\lambda > \mu$ we have a problem — the system will not reach equilibrium; we will have customers arriving faster than we can deal with them. Although we will not prove it, it is sufficient and necessary that $\lambda < \mu$ for the system to reach equilibrium.

At this point we have the probabilities for every possible state of the system. From these we can derive the properties we are interested in. For example what is the average number of customers, $\overline{N}$ in the system?

$$\begin{aligned} \overline{N} &= \sum_{k=0}^{\infty} k p_k \\ &= \sum_{k=0}^{\infty} k(1-\rho)\rho^k \\ &= (1-\rho)\rho\frac{\partial}{\partial\rho}\left(\sum_{k=0}^{\infty}\rho^k\right) \\ &= (1-\rho)\rho\frac{\partial}{\partial\rho}\left(\frac{1}{1-\rho}\right) \\ &= \frac{\rho}{1-\rho} \end{aligned}$$

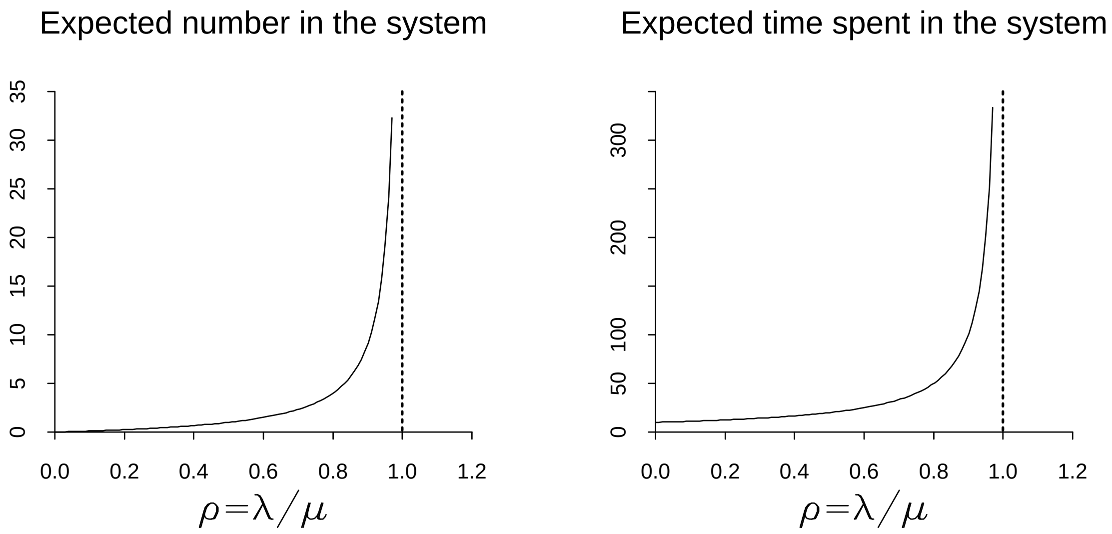


Figure 1: M/M/1 queueing system

How long, on average, will a customer spend in the queue? A customer arriving in the queue will find, on average $\overline{N}$ in the system, and will spend a time, say $\overline{T}$, in the system before departure. During that time there will be, on the average $\lambda\overline{T}$ arrivals which should leave $\overline{N}$ customers in the queue. In other words the customer leaves the queue as he finds it. Therefore,

$$\overline{N} = \lambda\overline{T}$$

which is known as Little's result. In our case

$$\begin{aligned} \overline{T} &= \frac{\overline{N}}{\lambda} \\ &= \frac{\frac{1}{\mu}}{1-\rho} \end{aligned}$$

which is the M/M/1 average response time. Note that

- μ is the service rate
- $\frac{1}{\mu}$ is the average service time
- ρ is the utilization

At low utilizations we expect the queue to be empty and the average response time becomes the average service time. At high utilizations, that is as ρ approaches one, the response time is unbounded. This is an extremely important thing to know and applies to queueing systems quite generally and is illustrated in Figure 1. Consider the effect on response time by increasing the utilization by a constant load of 0.1.

utilization (ρ)			response time ($\overline{T}$)		
old	new	% increase	old	new	% increase
0.1	0.2	100.0	$1.11\ \frac{1}{\mu}$	$1.25\ \frac{1}{\mu}$	13
0.5	0.6	20.0	$2\ \frac{1}{\mu}$	$2.5\ \frac{1}{\mu}$	25
0.8	0.9	12.5	$5\ \frac{1}{\mu}$	$10\ \frac{1}{\mu}$	100

Predicting response times is thus very difficult at high loads (although the unpredictability is easy to predict). *Running systems at maximum utilization may please the providers, but it doesn't please the users.*

2.5 Other Simple Queueing Systems

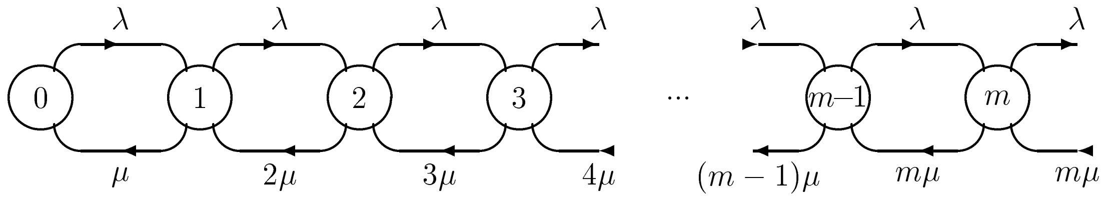
There are other systems which birth death systems can be used to model. We will not solve them completely here (the solutions tend to be messier) but just indicate how the birth and death rates can be modified to model other systems.

2.5.1 M/M/m — m servers

Consider the M/M/m system. This is just like the M/M/1 system, except that there are m servers in the system. As before, for all $k, \lambda_k = \lambda$, but now the service rate is a function of the number of customers in the system

$$\mu_k = \begin{cases} k\mu & \text{if } 0 < k \leq m \\ m\mu & \text{if } k \geq m \end{cases}$$

For an equilibrium distribution to exist we will need to require that $\frac{\lambda}{m\mu} < 1$.



The state diagram for the M/M/m queue is shown above. The service rate increases as the number of customers in the system increases until all servers are busy (at state m).

The same flow balance equations as in the M/M/1 case apply, although the μ_i are now not all equal.

M/M/m Performance Measures

The difference equations in this case are given by:

$$
\begin{aligned}
0 &= \mu p_1 - \lambda p_0 \\
0 &= (k+1)\mu p_{k+1} - (\lambda + k\mu)p_k + \lambda p_{k-1} \qquad \text{if } 0 < k \leq m \\
0 &= m\mu p_{k+1} - (\lambda + m\mu)p_k + \lambda p_{k-1} \qquad \text{if } k \geq m
\end{aligned}
$$

From which we may easily solve iteratively for the p_k as follows:

$$
\begin{aligned}
p_1 &= \lambda/\mu p_0 \\
p_2 &= 1/2(\lambda/\mu)^2 p_0 \\
&\vdots \\
p_m &= 1/m!(\lambda/\mu)^m p_0 \\
p_{m+1} &= 1/(mm!)(\lambda/\mu)^{m+1} p_0 \\
&\vdots
\end{aligned}
$$

This gives us the general expression for p_k:

$$
\begin{aligned}
p_k &= (\lambda/\mu)^k \frac{p_0}{k!} \qquad \text{if } k \leq m \\
&= (\lambda/\mu)^k \frac{p_0}{m!m^{k-m}} \qquad \text{if } k \geq m
\end{aligned}
$$

Recalling that we also require that $\sum_{k=0}^{\infty} p_k = 1$, we may solve for p_0:

$$
\begin{aligned}
p_0 &= \left[\sum_{k=0}^{m-1} \left(\frac{\lambda}{\mu}\right)^k \frac{1}{k!} + \frac{m^m}{m!} \sum_{k=m}^{\infty} \left(\frac{\lambda}{m\mu}\right)^k \right]^{-1} \\
&= \left[\sum_{k=0}^{m-1} \frac{(m\rho)^k}{k!} + \frac{(m\rho)^m}{(1-\rho)m!} \right]^{-1}
\end{aligned}
$$

The above makes use of the fact that $\rho = \frac{\lambda}{m\mu} < 1$, whence the second term in the equation for p_0 converges. In other words

$$
\begin{aligned}
\sum_{k=m}^{\infty} \rho^k &= \sum_{k=0}^{\infty} \rho^k - \sum_{k=0}^{m-1} \rho^k \\
&= \frac{1}{1-\rho} - \frac{1-\rho^m}{1-\rho} \\
&= \frac{\rho^m}{1-\rho}
\end{aligned}
$$

The performance measures for this queue can be obtained from the stationary solution, as before. In this case it is easiest to start with the derivation of $\bar{N}_q$ – the average number of customers waiting for service.

$$\begin{aligned}
\bar{N}_q &= \sum_{n=m+1}^{\infty} (n-m)p_n \\
&= \sum_{n=m+1}^{\infty} \frac{(n-m)m^m \rho^n p_0}{m!} \\
&= \frac{(m\rho)^m p_0}{m!} \sum_{n=m+1}^{\infty} (n-m)\rho^{n-m} \\
&= \frac{(m\rho)^m p_0}{m!} \rho \sum_{k=1}^{\infty} k\rho^{k-1} \\
&= \frac{(m\rho)^m p_0}{m!} \rho \frac{\partial}{\partial \rho} \left(\frac{1}{1-\rho} \right) \\
&= \frac{(m\rho)^m p_0 \rho}{m!(1-\rho)^2} \\
&= \frac{\rho p_m}{(1-\rho)^2}
\end{aligned}$$

With $\bar{N}_q$, we can obtain any of the other performance measures using Little's formula on the customers in the waiting line: $\bar{N}_q = \bar{T}_q \lambda$, and thence obtain performance measures for the whole queue using $\bar{T} = \bar{T}_q + \frac{1}{\mu}$.

2.5.2 M/M/1/K — finite queue capacity

Another interesting system is one with finite queue capacity, the M/M/1/K queueing system. Here if there are K customers in the system, any further arrivals are turned away. This is very much like an M/M/1 queueing system except that for $k \geq K$, $\lambda_k = 0$ and for $k > K$, $p_k = 0$. Using the equations from the M/M/1 queueing system, but limiting the summation, and again writing $\rho = \frac{\lambda}{\mu}$,

$$p_k = p_0 \rho^k \quad \text{for } k \leq K$$

and

$$\begin{aligned}
p_0 &= \frac{1}{1 + \sum_{k=1}^{K} \rho^k} \\
&= \frac{1}{1 + \frac{\rho - \rho^{K+1}}{1-\rho}} \\
&= \frac{1-\rho}{1-\rho^{K+1}}
\end{aligned}$$

which gives the result for $p_k, k \leq K$:

$$p_k = \frac{1-\rho}{1-\rho^{K+1}}\rho^k.$$

Note that, p_0, the probability that the system is idle, is greater than in the M/M/1 case. For this system with a finite state space an equilibrium distribution always exists whatever the arrival and departure rates.

2.5.3 M/M/1//N — finite customer population

Another interesting system is the finite customer population system, the M/M/1//N system. This has a single server, an unbounded queue capacity and a population of N customers. We can deal with this system by modifying the λ_k to model the arrival rate. Instead of having an arrival rate for the population as a whole, we can assign an arrival rate to each customer, say λ. If there are no customers in the system, then all of them are eligible to be born, so that

$$\lambda_0 = N\lambda\,.$$

As we have more customers in the system, we have less eligible to be born. So that,

$$\lambda_k = (N-k)\lambda, \quad \text{for } 0 \leq k \leq N\,.$$

With a single server the service rate is constant

$$\mu_k = \mu, \quad \text{for } k \geq 1\,.$$

Solving the difference equations for this queue we get

$$\begin{aligned} p_k &= p_0 \prod_{i=0}^{k-1} \frac{\lambda(N-i)}{\mu} \text{ for } 0 \leq k \leq N \\ &= \begin{cases} p_0\rho^k \dfrac{N!}{(N-k)!} & \text{if } 0 < k \leq N \\ 0 & \text{if } k > N \end{cases} \end{aligned}$$

and

$$p_0 = \left[\sum_{k=0}^{N} \rho^k \frac{N!}{(N-k)!}\right]^{-1}$$

2.5.4 M/M/m/m — m server loss system

Consider an M/M/m/m system in which there are m servers and in which customers will be turned way unless there is a free server (the system only has the capacity to hold m customers. An application of this system is to model a link in a telephone network. Such a link contains m circuits each of which is capable of carrying a single telephone call. Suppose that calls arrive at the link according to points from a Poisson process (ie the times between arrivals obey the memoryless property) of rate λ. If there is a free circuit the call is connected and holds the circuit for an exponentially distributed length of time, with mean $\frac{1}{\mu}$. At the end of this holding period the call terminates and the circuit is freed. If there are no free circuits then the call is lost.

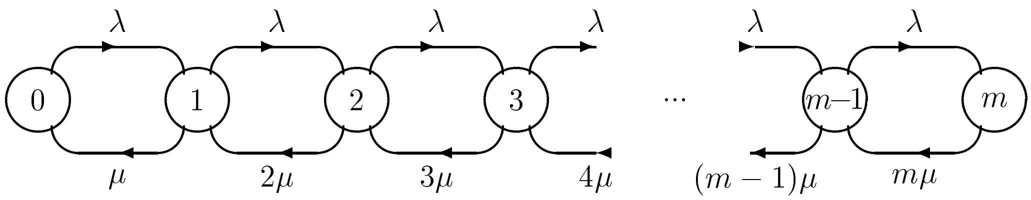
The birth and death rates are given as follows

$$\lambda_k = \begin{cases} \lambda & k < m \\ 0 & k \geq m \end{cases}$$

and

$$\mu_k = k\mu \quad \text{for } 1 \leq k \leq m\,.$$

The state transition diagram for this system is shown below.



The flow balance equations give for $k \leq m$

$$\begin{aligned} p_k &= p_0 \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}} \\ &= p_0 \prod_{i=0}^{k-1} \frac{\lambda}{(i+1)\mu} \\ &= p_0 \left(\frac{\lambda}{\mu}\right)^k \frac{1}{k!} \end{aligned}$$

and $p_k = 0$ for $k > m$. Solving for p_0 gives

$$\begin{aligned} \sum_{k=0}^{m} p_k &= 1 \\ p_0 \sum_{k=0}^{m} \left(\frac{\lambda}{\mu}\right)^k \frac{1}{k!} &= 1 \end{aligned}$$

so that

$$p_0 = \left[\sum_{k=0}^{m} \frac{\lambda}{\mu}^k \frac{1}{k!} \right]^{-1} .$$

The probability that an arriving call finds all circuits occupied, p_m, is called the *loss probability* for the telephone link. Thus,

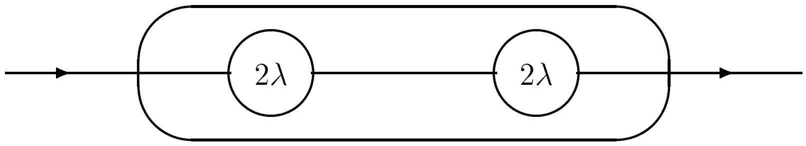
$$\begin{aligned} p_m &= p_0 \left(\frac{\lambda}{\mu}\right)^m \frac{1}{m!} \\ &= \left(\frac{\lambda}{\mu}\right)^m \frac{1}{m!} \left[\sum_{k=0}^{m} \left(\frac{\lambda}{\mu}\right)^k \frac{1}{k!} \right]^{-1} . \end{aligned}$$

This expression for the loss probability is known as Erlang's formula.

2.6 Extensions to Birth Death Models

The systems we have described so far are limited in that they only deal with exponential distributions. Also, customers queue for service at one of a number of identical servers, are serviced and then exit from the system. By extending birth death models we can model more general distributions and more complex networks of servers.

First we will turn to the arrival process distribution. How can we model a less variable arrival process, that is, one in which the coefficient of variation of the interarrival time is less than one? Consider a system in which a birth occurs in two stages. When the first half of an arrival occurs, the system enters a state in which no other arrival can begin until the current arrival is complete.



Each of these two stages in the birth will have exponentially distributed residence times. If the desired birth rate is λ, then let each stage have a rate 2λ. The average time to get through the combined birth process will be

$$\overline{\tau} = \frac{1}{2\lambda} + \frac{1}{2\lambda} = \frac{1}{\lambda} .$$

Since each stage has exponentially distributed residence times, the variance of each stage is

$$\sigma^2_{single} = \frac{1}{(2\lambda)^2} .$$

The two stages are independent, so the variance of τ, the time to get through both stages is

$$\sigma_\tau^2 = \frac{1}{(2\lambda)^2} + \frac{1}{(2\lambda)^2} = \frac{1}{2\lambda^2}$$

and so the coefficient of variation is

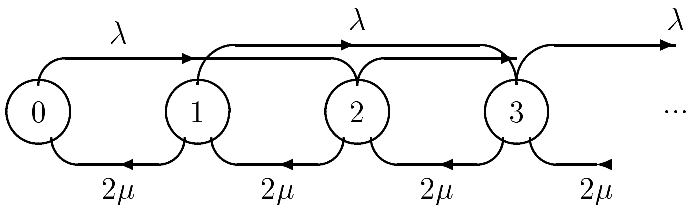
$$C_\tau = \frac{\sqrt{\frac{1}{2\lambda^2}}}{\frac{1}{\lambda}} = \frac{1}{\sqrt{2}}.$$

In general, if we use r stages each with rate $r\lambda$ we get an average time through all stages of $\frac{1}{\lambda}$ and a coefficient of variation of $\frac{1}{\sqrt{r}}$. The distribution that describes this process is the Erlangian distribution denoted E_r. We can use this method of stages either in the arrival process or in the server itself to decrease the coefficient of variation of the service times. Thus there are systems: $\mathrm{M}/\mathrm{E}_r/1$, $\mathrm{E}_r/\mathrm{M}/1$ and even $\mathrm{E}_r/\mathrm{E}_s/1$.

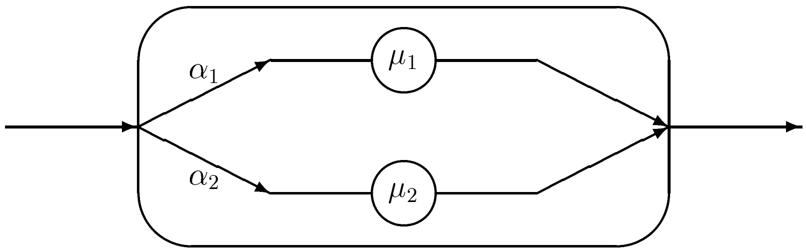
Example, $\mathbf{M/E_2/1}$

We can use a Markov process to model this two stage departure example by allowing the state of the process to represent the number of stages remaining to be served. An arrival increases the number of stages remaining to be served by two and occurs at rate λ while departures from individual stages reduces the number of stages to be served by one and occurs at rate 2μ.

The state transition diagram is shown below. The calculation of the equilibrium distribution is messy but can be done (see Kleinrock for details).



We have seen that combining stages in series reduces the coefficient of variation. If, instead, we combine them in parallel with a probability α_i of choosing the i^{th} parallel stage we can obtain a distribution with coefficient of variation larger than one.



It is shown in Kleinrock that if we combine r stages in parallel taking stage i with probability α_i and where the exponential service rate at stage i is μ_i then the coefficient of variation is given by

$$C_\tau = \frac{2\sum_{i=1}^{r}\frac{\alpha_i}{\mu_i^2}}{\left(\sum_{i=1}^{r}\frac{\alpha_i}{\mu_i}\right)^2} - 1 \geq 1\,.$$

These extensions required to model different types of distribution give rise to more complex state transitions diagrams and thus more complex systems of equations to solve, but in principle it can be done.

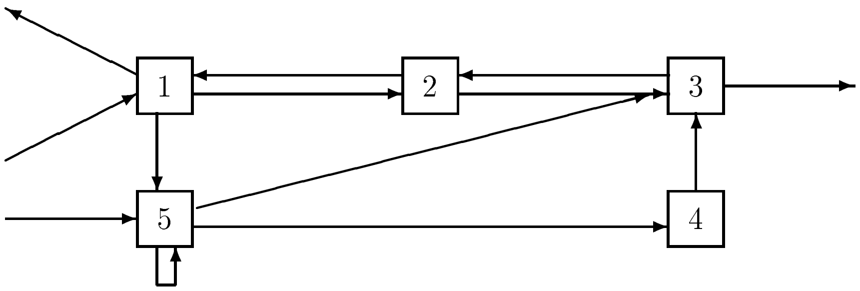
Another extension to birth death processes is to allow a number of independent servers. For example, if we have two servers, each with it's own queue, we can describe the state of the system by an ordered pair giving the number of customers at each server. In general, if we have n independent servers, we will end up with an n dimensional problem. Again in principle possible, but getting less attractive.

2.7 Queueing Networks

Some of the most interesting applications of probabilistic modelling techniques are in the area of distributed computing — various computational tasks are carried out at different locations.

A queueing network can be thought of as a connected (directed) graph with jobs requiring service at node i before moving (instantaneously) to node j.

For example,



2.7.1 Open Networks

We shall now consider *open* networks. Here there is at least one arc along which jobs enter the network and at least one arc along which jobs leave it and from every node it is possible to follow a path leading eventually out of the network. Hence, no job can be trapped and prevented from leaving.

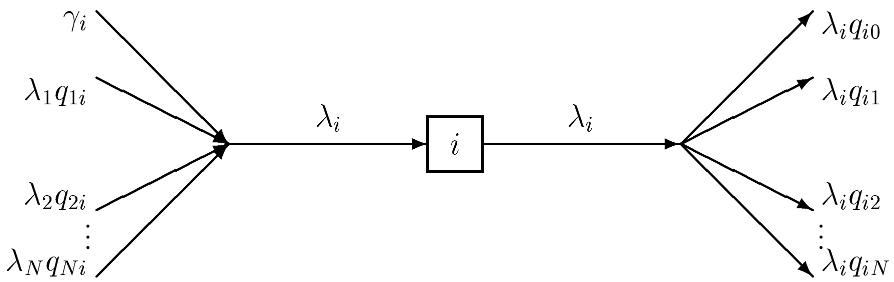
More formally, suppose we have a network consisting of N nodes, numbered $i = 1, 2, \ldots, N$. Node i contains a single server with unlimited waiting room. The service time of a job at node i is exponential, with mean $\frac{1}{\mu}$.

Jobs arrive from outside the network into node i in a Poisson stream with rate γ_i. Whenever a job completes service at node i it goes to node j with probability q_{ij} ($i, j = 1, 2, \ldots, N$; $\sum_{j=1}^{N} q_{ij} \leq 1$). A job leaves the network from node i with probability

$$q_{i0} = 1 - \sum_{j=1}^{N} q_{ij} \,.$$

The probabilities q_{ij} are called the *routing probabilities* of the network and the $N \times N$ matrix $Q = (q_{ij})$ its *routing matrix*. Such a model of an open network with parameters γ_i, μ_i and Q is called a *Jackson network*. The system state is described by the vector $(k_1, k_2, \ldots, k_N)$, where k_i is the number of jobs present at node i. Our assumptions ensure that the process is Markov. We consider its steady state behaviour.

Suppose that λ_j is the total average number of jobs arriving into (and leaving) node j per unit time. On average, λ_j jobs leave node j per unit time; of these, a fraction q_{ji} go to node i. Therefore, the rate of traffic from node j to node i is $\lambda_j q_{ji}$.



Adding together all contributions,

$$\lambda_i = \gamma_i + \sum_{j=1}^{N} \lambda_j q_{ji} \quad i = 1, 2, \ldots, N \,.$$

These are known as the *traffic equations*. It can be shown that, for open networks, the traffic equations determine the arrival rates λ_i uniquely. A necessary and sufficient condition for the existence of an equilibrium distribution is that

$$\rho_i \triangleq \frac{\lambda_i}{\mu_i} < 1$$

where λ_i is the solution of the traffic equations.

Let $p(k_1, k_2, \ldots, k_N)$ be the steady state equilibrium distribution. Then *Jackson's theorem* states that

$$p(k_1, k_2, \ldots, k_N) = p_1(k_1)p_2(k_2) \cdots p_N(k_N)$$

where $p_i(k_i)$ is the equilibrium distribution that there are k_i jobs in an M/M/1 queue with traffic intensity ρ_i.

Jackson's theorem has some important implications.

1. The numbers of jobs at the various nodes are independent.
2. Node i behaves as if subjected to a Poisson stream with rate λ_i.

Jackson's theorem may be generalized to the case where node i has n_i servers and so the nodes behave as independent M/M/n_i queues

2.7.2 Closed Networks

There are many systems where jobs can be delayed at a number of nodes, and where the total number of jobs available, or admitted, is kept constant over long periods of time. Such systems are modelled by *closed* networks — that is, where there are no external arrivals or departures.

Suppose that node i has a single server with service times being exponentially distributed with mean rate $\frac{1}{\mu_i}$. No jobs leave the network, so the routing probabilities satisfy

$$\sum_{j=1}^{N} q_{ij} = 1\,, \quad i = 1, 2, \ldots, N$$

There are no external arrivals so the number of jobs in the system is always a constant, denoted by K. The states of the system, described by the vector $(k_1, k_2, \ldots, k_N)$, thus satisfy the constraint,

$$\sum_{i=1}^{N} k_i = K\,.$$

The state space, S, is then finite. The number of states is

$$\binom{K + N - 1}{N - 1}$$

being the number of ways in which the integer K can be partitioned into N non-negative components where

$$\binom{n}{r} \triangleq \frac{n!}{(n-r)!r!}\,.$$

The traffic equations become

$$\lambda_i = \sum_{j=1}^{N} \lambda_j q_{ji}\,, \quad i = 1, 2, \ldots, N\,.$$

Again, with a finite state space there always exists an equilibrium distribution whatever the service rates μ_i and arrival rates λ_i.

Analogous to Jackson's theorem for the open network case it may be shown that

$$p(k_1, k_2, \ldots, k_N) = \frac{1}{G} r_1(k_1) r_2(k_2) \cdots r_N(k_N)$$

where $r_i(k_i)$ is the probability that there are k_i jobs in an M/M/1 queue with traffic intensity given by a solution to the traffic equations. The situation with a closed network differs from that in an open network in that the normalization constant G has to be determined by

$$G = \sum_{s \in S} r_1(k_1) r_2(k_2) \cdots r_N(k_N)$$

obtained by summing over all states $s = (k_1, k_2, \ldots, k_n)$ in the state space S.

Several differences exist between the cases of open and closed queueing networks. First, there is the need with closed networks to compute the normalization constant G — a computationally hard problem involving the summation over a (large) state space, S. Secondly, the presence of the constraint $\sum_{i=1}^{N} k_i = K$ means that the numbers of jobs in the various queues are no longer independent. For instance, consider the extreme case where all K jobs are at one node — then all other nodes must be empty since there is a constant number of jobs, K, in the network at all times.

3 The M/G/1 Queue

Our detailed analysis of the M/M/1 queueing system was made possible by the simplifying assumption of independent exponential interarrival and service times. In this case the process $X(t)$ describing the number in the system has the following extremely important property: At any given time t_0 the future behavior of $X(t)$ depends only on the current state $X(t_0)$ and not on the past prior to t_0. This memoryless property defines Markov processes, and it introduces a substantial simplification into the analysis. Many real queueing systems, however, do not conform to these assumptions. Can we honestly convince ourselves that the transmission time for IP traffic at a router conforms to an exponential distribution? Even the Markovian M/H_2/1 queue might be unable to model the service process in this case.

To solve models for which the Poisson assumption is clearly inappropriate we need to study a more general formulation of the queueing model. In this section we study the M/G/1 queue – here we will retain the Poisson assumption for the arrivals process but will be interested in carrying out the analysis of $X(t)$ assuming general service time distributions. With this assumption, if at some point t_0 a job is in execution, then the behavior of $X(t), t > 0$, will depend not only on $X(t_0)$ but on the amount of service already received by the executing job. Thus $X(t)$ is no longer a Markov process.

An extension to our previous modelling approach which is suggested by the above property can be sketched as follows. We can redefine $X(t)$ so that its state space includes the values of a supplementary variable that x that specifies the amount of service already received by the job (if any) in execution at time t. Specifically, the value of the new process $X'(t)$ is a pair (n, x), where n is the number in the system at time t and x is defined as previously. The new process now has the Markov property and the analysis can proceed on that basis.

We shall avoid the difficult analysis of this approach and consider an alternative which is far simpler for our limited purposes. The technique is motivated by recognition of the fact that although $X(t)$ does not have the Markov property at every point in time there exist many embedded sequences of time points at which $X(t)$ does have the Markov property. Suppose, for example, that we define the time instants (or epochs) $t_1, t_2, \ldots$ so that t_i is the instant immediately after the i^{th}job execution in an M/G/1 system with service distribution $B(s)$.

Since the service and interarrival intervals are mutually independent, $X(t_i)$ at the epochs t_i has the Markov property. As a result the t_i are also called regeneration points. The discrete time process $X(t_i), i = 1, 2 \ldots,$ constitutes a Markov chain which is said to be an *embedded* Markov chain with respect to $X(t)$.

The solution of the M/G/1 model is difficult, and involves the use of Laplace Transforms and generating functions. Rather than devote excessive time to the derivation of the results, we simply present an intuitive derivation which allows us to derive the relevant performance measures for this system.

3.1 Performance Measures

Let $\bar{s}$ denote the mean service time of a customer in the M/G/1 queue. We begin with an intuitive derivation of the queueing time of a customer before it receives service. We know that this is given by the simple relationship

$$\bar{T}_q = \bar{N}_q \bar{s} + \bar{x},$$

where $\bar{x}$ is the remaining service time of the customer in service at the arrival of the given customer. It turns out that $\bar{x} = \frac{1}{2}\lambda\bar{s^2}$ (using arguments from renewal theory). We also know that

$$\bar{N}_q = \lambda \bar{T}_q$$

from Little's law. Substituting and solving for $\bar{T}_q$ gives the solution:

$$\bar{T}_q = \frac{\lambda \bar{s^2}}{2(1-\lambda\bar{s})}.$$

Writing $\rho = \lambda\bar{s}$, and C_s for the coefficient of variation of the service time gives us the solution we are after:

$$\bar{T}_q = \frac{\bar{s}\rho(1+C_s^2)}{2(1-\rho)} \tag{3}$$

From this we may derive directly the formula for $\bar{T}$, the mean response time

$$\bar{T} = \bar{s}\left[1 + \frac{\rho(1+C_s^2)}{2(1-\rho)}\right]. \tag{4}$$

This widely known formula is called the *Pollaczek-Khintchine* or PK formula. From Little's formula, equations for the other performance measures follow directly. For example, the expected queue length is given by:

$$\bar{N} = \rho + \frac{\rho^2[1+C_s^2]}{2(1-\rho)}. \tag{5}$$

According to the definition of the embedded Markov chain, Equation (5) represents the mean number of jobs that are left in the system by departing jobs. It is important to realise that since we observe the system only at departure times, during the service time of the departing customer more than one new arrival may have occured. You might like to construct the Markov chain for this model and verify for yourself by substitution of the appropriate parameters into Equation (5) that we obtain the earlier results for the M/M/1 queue.

Note particularly that $\bar{N}$ is determined not only by the mean interarrival and service times but also by the second moment of the service time distribution. As shown by Equation (5),

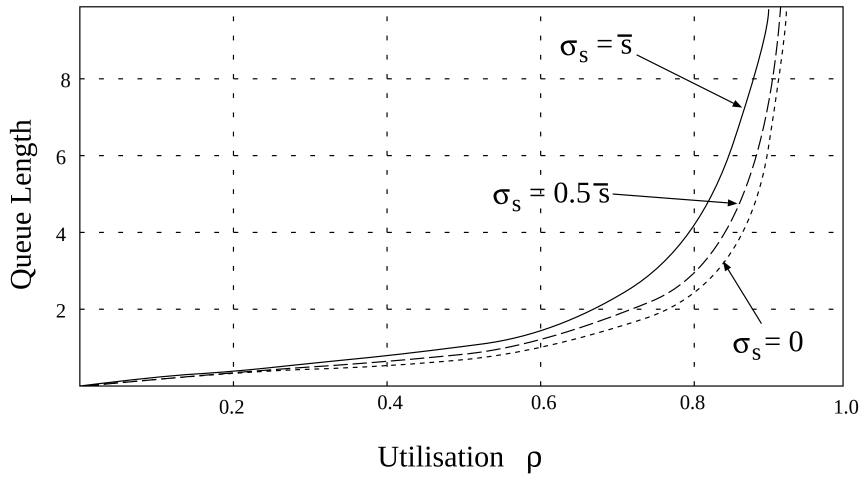


Figure 2: M/G/1 queue length depends on first 2 moments of service time distribution

$\bar{N}$ increases linearly with the variance of the service times. Considering constant service times, for example, the variation in service times is zero and we have

$$\bar{N} = \rho + \frac{\rho^2}{2(1-\rho)} = \frac{\rho(1-\rho/2)}{(1-\rho)} \tag{6}$$

which is to be compared with the expression $\rho/(1-\rho)$ for exponential service times (whose coefficient of variation is 1). For small ρ the difference in the means is small, but as ρ approaches 1 (the saturation condition), the two differ by almost a factor of 2. Figure 2 shows the queue length for this system for a range of values of the coefficient of variation of the service time. Note that the poorest performance is exhibited by the exponential service time and the best by the deterministic or constant service time. This confirms our earlier remarks that the exponential service time distribution can frequently be used as a worst-case in queueing systems analysis.

Since both the first and second moments of the service time distribution are important, the coefficient of variation can be used to determine the behaviour of the system. What values of $C_s = \sigma_s/\bar{s}$ is one likely to encounter? We can usefully consider 4 regions:

- zero: This is the case of constant service times, for example ATM networks where all cells have the same fixed size.
- ratio < 1: Since this ratio is better than the exponential case, using an M/M/1 model will give performance measures which are conservative (i.e. safe).
- ratio close to 1: This is the most common occurrence, and corresponds to exponential service time. Many systems correspond to this model, for example packet switched networks with variable size packets, batch jobs on a mainframe etc.
- ratio > 1: If you observe this then you need to use an M/G/1 model. The most common occurrence of this in a communications system is a bimodal distribution,

with a wide spread between the peaks. A good example is Internet traffic – many small packets (telnet connections), and the bulk of the remainder large, but centered around the MTU.

3.2 What about non-Poisson arrivals ?

Thus far we have only a the arrivals process to be Poisson. The same considerations apply to the arrival rate – for a Poisson arrival stream the ratio of standard deviation to mean is 1, and if the observed ratio is much less than 1, then arrivals tend to the evenly spaced. If the ratio is greater than 1, arrivals tend to cluster and congestion becomes more acute. In general, for the same mean arrival rate, the performance of any queueing system in which arrivals are completely random will be worse than a system in which arrivals are completely predictable. Intuitively this makes sense – in the former batches of arrivals may occur within a short period, causing congestion, whereas in the latter this is less likely to occur. A coefficient of variation C_a between 0 and 1 exhibits properties between those of a constant and Poisson distribution. Leung gives the following rules of thumb

- $1 \leq C_a \leq 1.15$ – Poisson is a reasonable approximation
- $0.85 \leq C_a$ – Process becomes more constant
- $C_a \geq 1.15$ – Arrivals tend to be bunched.

Note that the Poisson assumption tends to over-estimate congestion for processes with $C_a \leq 1$, and under-estimate for $C_a > 1$.

Example: An IP service provider has a bank of dial-in modems, each of which recieves messages according to a Poisson distribution with rate $\lambda_j = 10j$ packets per second, for $j = 1 \ldots 5$. Messages are multiplexed onto a single output link and transmitted in arrival order. Calculate the mean datagram response time and the mean number in the system if IP packets have a bimodal size distribution which gives them a mean service time of 5ms with a variance of 4ms.

3.3 A word on applicability

The M/G/1 queueing model is widely applicable – in fact it will apply to any system with a Poisson arrivals process and a service discipline in which :

- *work conserving* – work is neither created nor destroyed (customers cannot die/leave the queue, split in two etc),
- the service selection mechanism is not based on the *service time* of a particular customer, and
- scheduling of customers is non-preemptive.

For any system which obeys these constraints, the PK formula applies. It should be pointed out though that the invariance in the response time to the scheduling rule applies to the *mean value only*, and not to the distribution or its other moments. In the case of the number in the system, however, the invariance does apply to the distribution and the higher moments.

4 $\mathrm{M}_i/\mathrm{G}_i/1$ Non-preemptive Service Priority

This is the final single queue system which we will study. Far more complicated queueing and service disciplines have been analysed, but the derivation of the performance measures becomes significantly more difficult with additional complexity. We will look at the case where a single server queue serves 2 priority classes, as shown in Figure 3.

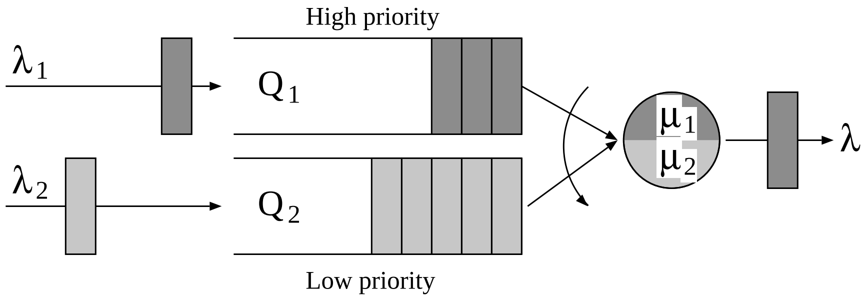


Figure 3: The $\mathrm{M}_i/\mathrm{G}_i/1$ Non-preemptive service priority queue

The key to the analysis of this system is that the service is non-preemptive. A low priority customer cannot be interrupted by the arrival of a high priority customer, and will complete service before the high priority customer is served. Denote the two service queues by $Q_i, i = 1, 2$, where Q_1 is the high priority queue. The arrivals processes to Q_i are Poisson distributed at rates λ_i and the service processes are of general distribution with means $\bar{s_i} = 1/\mu_i$ and variances $\sigma^2_{s_i}, i = 1, 2$.

The expected service time of the average customer is

$$\bar{s} = \frac{\lambda_1}{\lambda\mu_1} + \frac{\lambda_2}{\lambda\mu_2}$$

which follows directly from the definition.

We commence with the derivation of the average time spent in the queue:

$$\bar{T}_q = \frac{\lambda_1}{\lambda}\bar{T}_{q_1} + \frac{\lambda_2}{\lambda}\bar{T}_{q_2}$$

where

$$\bar{T}_{q_1} = \mathbb{E}(Y_0) + \mathbb{E}(Y_1)$$

and

$$\begin{aligned} Y_0 &= \text{remaining service time of the customer in service} \\ Y_1 &= \text{service time of the customers ahead of the arrival} \end{aligned}$$

Thus, an arriving customer sees the sum of the remaining service of the customer currently receiving service plus the service time of the customers ahead of it in the queue.

Clearly

$$\mathbb{E}(Y_1) = \frac{\bar{N}_{q_1}}{\mu_1} = \frac{\lambda_1 \bar{T}_{q_1}}{\mu_1} = \rho_1 \bar{T}_{q_1}$$

where we have defined $\rho_1 = \lambda_1/\mu_1$. Thus,

$$\bar{T}_{q_1} = \frac{\mathbb{E}(Y_0)}{1-\rho_1}.$$

Similarly, for arrivals to the second queue Q_2,

$$\bar{T}_{q_2} = \mathbb{E}(Y_0) + \mathbb{E}(Y_1) + \mathbb{E}(Y_2) + \mathbb{E}(Y_{1'})$$

where

$$\begin{aligned} Y_2 &= \text{service time of customers in } Q_2 \\ Y_{1'} &= \text{service time of customers joining } Q_1 \text{ during the wait at } Q_2 \end{aligned}$$

So

$$\mathbb{E}(Y_2) = \rho_2 \bar{T}_{q_2}$$

and

$$\begin{aligned} \mathbb{E}(Y_{1'}) &= \mathbb{E}(\text{number of arrivals at } Q_1 \text{ during service of } Q_2) \\ &\times \mathbb{E}(\text{service time of a customer in } Q_1) \\ &= \lambda_1 \bar{T}_{q_2} \times 1/\mu_1 \end{aligned}$$

Putting this all together gives

$$\begin{aligned} \bar{T}_{q_2} &= \mathbb{E}(Y_0) + \rho_1 \bar{T}_{q_1} + \rho_2 \bar{T}_{q_2} + \rho_1 \bar{T}_{q_1} \\ &= \frac{\mathbb{E}(Y_0)}{(1-\rho_1)(1-\rho_1-\rho_2)} \end{aligned}$$

where $\rho_1 + \rho_2 < 1$.

In a similar fashion to the derivation of the remaining time of a customer in service at the M/G/1 queue, we can derive the following formula for $\mathbb{E}(Y_0)$:

$$\mathbb{E}(Y_0) = \frac{1}{2} \sum_{i=1}^{2} \lambda_i \bar{s_i^2},$$

which permits the computation of $\bar{T}_q$. Using Little's law, but restricting to each queue:

$$\begin{aligned}
\bar{N}_{q_i} &= \lambda_i \bar{T}_{q_i} \\
\bar{N}_q &= \lambda \sum_{i=1}^{2} \frac{\lambda_i}{\lambda} \bar{T}_{q_i} \\
&= \sum_{i=1}^{2} \bar{N}_{q_i} \\
\bar{T}_i &= \bar{T}_{q_i} + 1/\mu_i \\
\bar{N}_i &= \bar{N}_{q_i} + \rho_i \\
\bar{N} &= \sum_i \bar{N}_i
\end{aligned}$$

These results may be generalized for arbitrary numbers of queues Q_i – see Leung or Kleinrock for the full details.

Example: A second IP service provider who we are considering decides to queue the arriving traffic into two queues, on the basis of their sizes. Small datagrams take 2ms to transmit, on average, with standard deviation 2ms, and are given priority. Large datagrams take an average of 17ms to transmit, with standard deviation 2ms, and are given low priority. The arrivals process is as described previously. Suppose that small packets make up 80% of the traffic. Compare the performance of this system to one in which the priorities are reversed. Should we use this IP service provider (and why)?