10 Solving Queueing Models

10.1 Introduction

In this note we look at the solution of systems of queues, starting with simple isolated queues. The benefits of using predefined, easily classified queues will become apparent: many performance measures can be calculated directly from the parameters of the model. Obviously the situation becomes more complicated when queues are connected together. However we see that even in this case deriving performance measures can be very straightforward as we are able to consider each queue in isolation. Finally we summarise the assumptions which we need to make in order to obtain these solutions.

10.2 Single Queues

Since we assume that all the queues which we consider have a Markovian arrival process and a Markovian service process the queue can be modelled as a Markov process. The state transition diagram for a single-server queue with infinite capacity is shown in Figure 22. Note that unlike the Markov processes which we have considered earlier in the course this process has an infinite state space.

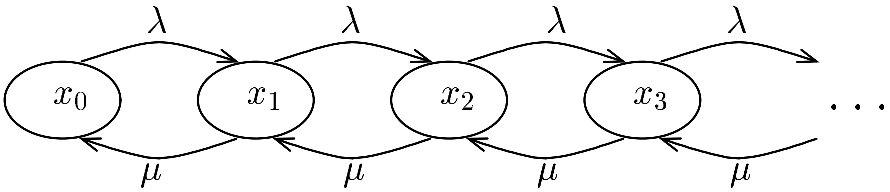

Figure 22: The state transition diagram for a simple $M/M/1$ queue

If we write the global balance equations for this system we can soon recognise a regular pattern emerging.

$$
\begin{aligned}
\lambda \boldsymbol{\pi_0} &= \mu \boldsymbol{\pi_1} \\
(\lambda + \mu)\boldsymbol{\pi_1} &= \lambda \boldsymbol{\pi_0} + \mu \boldsymbol{\pi_2} \\
(\lambda + \mu)\boldsymbol{\pi_2} &= \lambda \boldsymbol{\pi_1} + \mu \boldsymbol{\pi_3} \\
&\vdots
\end{aligned}
$$

Using simple algebra we can rewrite these as shown below:

$$
\begin{aligned}
\boldsymbol{\pi_1} &= \frac{\lambda}{\mu}\boldsymbol{\pi_0} \\
\boldsymbol{\pi_2} &= \frac{\lambda}{\mu}\boldsymbol{\pi_1} = \left(\frac{\lambda}{\mu}\right)^2 \boldsymbol{\pi_0} \\
\boldsymbol{\pi_3} &= \frac{\lambda}{\mu}\boldsymbol{\pi_2} = \left(\frac{\lambda}{\mu}\right)^3 \boldsymbol{\pi_0} \\
&\vdots
\end{aligned}
$$

Remembering that *traffic intensity*, ρ is defined to be λ/μ (for the single server case, $\lambda/(c \times \mu)$ in the general case) we can see that for an arbitrary state x_i

$$\boldsymbol{\pi_i} = \rho^i \boldsymbol{\pi_0}$$

and using the normalisation condition we see that

$$1 = \sum_{i=0}^{\infty} \boldsymbol{\pi_i} = \boldsymbol{\pi_0} \sum_{i=0}^{\infty} \rho^i = \boldsymbol{\pi_0} \frac{1}{1-\rho} \qquad \text{if } \rho < 1$$

From this we can deduce that

$$\boxed{\boldsymbol{\pi_0} = 1 - \rho \qquad \text{and} \qquad \boldsymbol{\pi_i} = (1-\rho)\rho^i \qquad \text{for all } i > 0.}$$

Thus we have a symbolic evaluation of the steady state distribution for any $M/M/1$ queue, i.e. the steady state distribution expressed in terms of the parameters of the model, λ and μ. This means that for any particular model we can find the steady state distribution simply by evaluating the expressions above with our particular value of $\rho = \lambda/\mu$. More importantly, we can derive symbolic expressions for the important performance measures we are likely to want to derive from a queue, and then evaluate those measures for a particular model *without having to carry out any numerical solution of global balance equations.*

Utilisation, U The queue is being utilised whenever it is non-empty; in other words the utilisation, U, is $1-\boldsymbol{\pi_0}$.

$$\boxed{\textbf{Utilisation} \qquad U = \rho}$$

Mean number of customers in the queue, N This is the expectation of the number of customers in the service facility as a whole, i.e.

$$N = \sum_{n=1}^{\infty} n \times \boldsymbol{\pi_n} = \sum_{n=1}^{\infty} n \times (1-\rho)\rho^n = \frac{\rho}{(1-\rho)}$$

$$\boxed{\textbf{No. in queue} \qquad N = \frac{\rho}{1-\rho}}$$

Mean number of customers waiting, N_b This is the expectation of the number of customers in the buffer i.e.

$$N_b = \sum_{n=1}^{\infty} (n-1) \times \boldsymbol{\pi_n} = \sum_{n=1}^{\infty} (n-1) \times (1-\rho)\rho^n = \rho \sum_{n=1}^{\infty} n \times (1-\rho)\rho^n = \frac{\rho^2}{(1-\rho)}$$

$$\boxed{\textbf{No. in buffer} \qquad N_b = \frac{\rho^2}{1-\rho}}$$

Mean response time, R Using Little's Law we can calculate the mean response time of the queue to be the mean number in the queue N, divided by the arrival rate λ, i.e. $R = N/\lambda = \frac{\rho}{1-\rho} \times \frac{1}{\lambda} = \frac{1/\mu}{1-\rho} = \frac{1}{\mu(1-\rho)}$.

$$\boxed{\textbf{Response time} \quad R = \frac{1}{\mu(1-\rho)}}$$

Other common performance measures can be calculated in a similar way. Similarly we can derive symbolic steady state distributions, and expressions for performance measures, for the other standard queues. Thus, given almost any single queue model, in order to derive a performance measure it is only necessary to select the appropriate formula from a table and evaluate it using the parameters of your model. Some examples are shown in Table 2. Most textbooks on performance models will contain these formulae.

Performance Measures	$M/M/1$	$M/M/c$	$M/M/1/K$
Traffic Intensity ρ	$\frac{\lambda}{\mu}$	$\frac{\lambda}{c \times \mu}$	$\frac{\lambda}{\mu}$
Utilisation U (per server)	ρ	ρ	$\rho(1 - \frac{(1-\rho)\rho^K}{1-\rho^{K+1}})$
Prob. system is idle $\boldsymbol{\pi_0}$	$1-\rho$	$\left(1 + \frac{(c\rho)^c}{c!(1-\rho)} + \sum_{n=1}^{c-1} \frac{(c\rho)^n}{n!}\right)^{-1}$	$\frac{1-\rho}{1-\rho^{K+1}}$
Prob. buffer non-empty B	ρ^2	$\frac{(c\rho)^c}{c!(1-\rho)}\boldsymbol{\pi_0}$	
Mean no. in system N	$\frac{\rho}{1-\rho}$	$c\rho + \rho B/(1-\rho)$	$\frac{\rho}{1-\rho} - \frac{(K+1)\rho^{K+1}}{1-\rho^{K+1}}$
Mean no. in buffer N_b	$\frac{\rho^2}{1-\rho}$	$\rho B/(1-\rho)$	$\frac{\rho}{1-\rho} - \rho\frac{1+K\rho^K}{1-\rho^{K+1}}$
Mean response time R	$\frac{1}{\mu(1-\rho)}$	$\frac{1}{\mu}\left(1 + \frac{B}{c(1-\rho)}\right)$	$\frac{N}{\lambda(1 - \frac{(1-\rho)\rho^K}{1-\rho^{K+1}})}$

Table 2: Common performance measures

Example: Consider again the network gateway discussed in the previous note. Measurements have shown that packets arrive at a mean rate of 125 packets per second, and are buffered until they can be transmitted. The gateway takes 2 milliseconds on average to transmit a packet. The gateway currently has 13 places (including the packet being transmitted) and packets that arrive when the buffer is full are lost. We wish to find out if the buffer capacity is sufficient to ensure that less than one packet per million gets lost.

We represent the gateway as a $M/M/1/13$ queue, with $\lambda = 125$ and $\mu = 1/0.002 = 500$. The utilisation of the gateway will be $\rho = \lambda/\mu = 0.25$.
The loss rate is the arrival rate multiplied by the probability that the system is full, i.e. $\lambda \times \boldsymbol{\pi_K}$. The proportion of packets that are lost is simply $\boldsymbol{\pi_K}$ (expected number lost per time unit divided by the expected number arriving per time unit)

$$\boldsymbol{\pi_K} = \frac{(1-\rho)\rho^K}{1-\rho^{K+1}} = (0.75\times 0.25^{13})/(1-0.25^{14}) = (1.1176\times 10^{-8})/0.99999999 = 1.12\times 10^{-8}$$

Thus the expected proportion of packets lost is 0.0112 every million packets, well within the requirement.

Exercise: Using maple or otherwise, investigate how much the buffer size can be reduced before the loss rate becomes unacceptable. What happens if the arrive rate increases to 175 packets per second?

10.3 Product Form Queueing Networks

Once we consider a network of queues rather than a single queue the possible forms of the state spaces of the underlying Markov process become much more diverse. Therefore we would not anticipate being able to derive symbolic steady state distributions of wide applicability in the same way as we have done for single queues. However, for a large class of queueing networks a straightforward and efficient means of solving models has been found. These networks are known as *product form networks.*

The term *product form* comes from the fact that the steady state distribution of these models can be derived as the product of the steady state distributions of each of the constituent service centres.

Recall that the state of a single service facility can be characterised by the number of customers currently in the system. In a queueing network the state of the system is characterised by the number of customers waiting at each of the service centres. This is usually represented as a tuple. For example, in a simple queueing network with two service centres, such as the one shown in Figure 23, the state $\boldsymbol{(n_1, n_2)}$ indicates that there are n_1 customers in service centre 1 (queueing or in service) and n_2 customers in service centre 2.

For a model such as this one to have a product form steady state distribution means that the distribution can be expressed as a product of terms representing the steady states of each of the service centres considered in isolation, e.g.

$$\boldsymbol{\pi}(n_1, n_2) = \boldsymbol{\pi_1}(n_1) \times \boldsymbol{\pi_2}(n_2)$$

In terms of the system what this means is that the two service centres reach their equilibrium behaviours and then behave independently of each other. (Recall that a probability

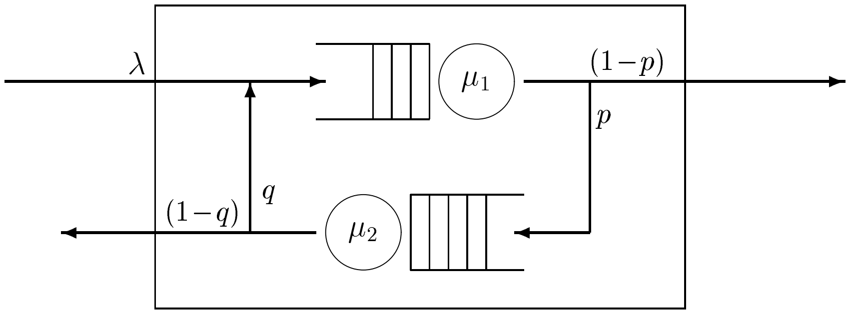

Figure 23: Open Queueing Network

of a coincidence of events can be expressed as the product of the probabilities of each event if and only if those events are independent.)

The class of queueing networks in which service centres exhibit this form of independent behaviour in equilibrium excludes some interesting and important system features as was discussed in the previous lecture note. However when the necessary conditions are satisfied, performance measures can be derived without resorting to the underlying Markov process—just as in the case of a single queue, because in effect we only solve the network one queue at a time.

10.3.1 Traffic equations

Using the already established formulae for individual service centres we only need to know two parameters: the arrival rate at the queue, which we will denote by λ_i for the ith queue, and the service rate at the queue, denoted μ_i. For a queueing network we will be given μ_i for each queue in the network. However, we will only usually be given the arrival rate of arrivals to the network from the environment (λ in the example shown in Figure 23). Fortunately the following theorems, the decomposition principle of Poisson/exponential distributions, and simple algebra, can help us to work out the arrival rate at each node in the network.

Burke's Theorem stated that a Poisson arrival process at a service centre with exponential service rates generates a Poisson process of departures. Moreover the rate of departure is the same as the arrival rate. This result implies that each service centre in a chain of simple exponential service centres with Poisson arrivals can be analysed independently using results from simple queues.

Jackson's Theorem generalised Burke's Theorem to consider an arbitrary network of exponential service centres each of which is driven by a Poisson arrival process. Although the presence of feedback paths destroys the Poisson nature of the service centre arrivals, Jackson showed that they still *behave* as if they were Poisson driven. Moreover the relationship between the arrival stream and the output stream is maintained. The networks considered by Jackson were *open*.

Again, this implies that each service centre in the queueing network can be analysed independently.

Gordon and Newell's Theorem considered a modification of Jackson's networks in which the network is *closed* rather than open. Closed queueing networks are not driven by any external Poisson arrival process. Instead, the number of customers in the network is fixed, K. Whenever a customer departs one service centre in the network it immediately requests service from another one. Hence customers never leave the network. Gordon and Newell's product form equation has the form:

$$\boldsymbol{\pi}(n_1, n_2, \ldots, n_k) = \frac{1}{G(K)} \prod_{i=1}^{n} \boldsymbol{\pi_i}(n_i)$$

where $G(K)$ is a *normalisation constant* chosen to ensure that the steady state probabilities sum to 1.

Using these results we know that if the arrival rate at queue i is λ_i then the departure rate will also be λ_i. Therefore, if all the departures from queue i go directly to queue $i+1$ the arrival rate at queue $i+1$ will also be λ_i, i.e. $\lambda_{i+1} = \lambda_i$.

However, in general the Poisson departure stream from a queue may be split between different queues and/or the environment, according to the routing probabilities. By the decomposition principle we know that when the departure stream is split in this way each of the resulting arrival streams will also be Poisson, with a rate obtained by multiplying the departure rate by the probability of this branch of the routing probability.

If we analyse all the service centres of a queueing network in this way, expressing its input stream as a sum of output streams from the environment or other service centres, we obtain what are known as the *traffic equations*. For the simple network shown in Figure 23, the traffic equations are as follows:

$$\begin{aligned} \lambda_1 &= \lambda + q \times \lambda_2 \\ \lambda_2 &= p \times \lambda_1 \end{aligned}$$

We obtain one equation for each service centre, and we have one unknown for each service centre, so we end up with n equations in n unknowns. Therefore we can solve the traffic equations and find the arrival rate at each service centre. In the case above, assume that $\lambda = 10$, $p = 0.5$ and $q = 0.4$. If we substitute $\lambda_2 = 0.5\lambda_1$ into the first equation we obtain

$$\begin{aligned} \lambda_1 &= 10 + 0.2\lambda_1 \\ 0.8\lambda_1 &= 10 \end{aligned}$$

Therefore $\lambda_1 = 12.5$ and substituting back we deduce that $\lambda_2 = 6.25$.

The great advantage of solving traffic equations rather than the global balance equations is that the number of equations we need to solve grows linearly with the number of service centres, rather than exponentially, which is the case for global balance equations.

10.4 Assumptions

Queueing network modelling is inherently a *top-down* process. The underlying philosophy is to begin by identifying the principal components of the system and the ways in which they interact, then supply any details that prove to be necessary. This philosophy inevitably leads to a large number of assumptions being introduced while a modelling study is being conducted.

Many of these assumptions are the general ones which we have been making throughout our study of Markov processes, although since we are avoiding the numerical solution of the global balance equations we *can* now consider models which have an infinite number of states. There are five assumptions which we must make about the behaviour of the model to ensure that a product form solution exists. These are:

Each service centre is **flow balanced**; this means that the number of customers that arrive at each centre is equal to the number of customers who depart.

The system exhibits **one step behaviour**; this means that no two customers in the system "change state" at exactly the same time. Here changing state may mean a customer finishing at one service centre and progressing to the next, or a customer arriving at, or departing from the system.

The system has **routing homogeneity**. This means that the probability that a customer completing service at service centre j now proceeds to service centre k is independent of the current queue lengths at j or k, for all service centres in the network.

The system has **device homogeneity**. The rate of completions of customers from a service centre may vary with the number of jobs at that centre, but otherwise cannot depend on the number or placement of customers within the network.

The system experiences **homogeneous external arrivals**; that is, the times at which arrivals from outside the network occur may not depend on the number or placement of customers within the network.