M/G/1 Priority Queues

Robert Marshall

Department of Mathematics and Statistics
The University of Melbourne

Master of Science Thesis
October 2012
Abstract

We examine an M/G/1 queue with several queueing disciplines. After reviewing some background material in probability theory, we consider a first-come first-served queue. Next, we examine an absolute priority queue, where high priority customers are always selected for service over lower priority customers, regardless of how long the latter have been waiting. We also review a method for calculating the expected waiting times for a variety of queueing disciplines, particularly an accumulating priority queue, where a customer’s priority is proportional to the time they have spent waiting. We then consider a newer method for determining the full waiting time distribution of an accumulating priority queue. Each of these distributions are given as functional relations for their Laplace-Stieltjes transforms, which in generally cannot be solved analytically. Thus, we conclude by demonstrating how to numerically invert these transforms, showing the waiting time distribution of an accumulating priority queue.
# Contents

1 Introduction ................................................................. 3  
1.1 Queueing Models ....................................................... 3  
1.2 Overview .................................................................. 4  

2 Preliminaries ................................................................. 6  
2.1 Laplace-Stieltjes Transforms ......................................... 6  
2.2 Wald's Identity ......................................................... 8  
2.3 Branching processes ................................................... 9  
2.4 Renewal Theory .......................................................... 11  
2.4.1 Limiting distribution of residual lifetime ..................... 13  
2.4.2 Joint distributions ................................................ 21  
2.5 Little's Law .............................................................. 22  

3 First-come, first-served M/G/1 queue .............................. 27  
3.1 Derived processes ....................................................... 28  
3.1.1 Virtual Work ...................................................... 28  
3.1.2 Maximum waiting time .......................................... 32  
3.2 Busy and idle periods .................................................. 32  
3.3 Waiting times ............................................................ 35  

4 Waiting time distribution in an M/G/1 queue with absolute priorities ........................................... 43  
4.1 Delay cycles .............................................................. 45  
4.1.1 Distribution of delay cycle length ............................... 47  
4.1.2 Distribution of waiting time for arrivals in a delay cycle ... 50  
4.2 Classifying queue states ............................................... 55  
4.2.1 Blocking times and initial delay durations ................. 57  
4.2.2 Delay cycle properties ......................................... 58  
4.2.3 Distribution of queue states .................................... 59  
4.2.4 Unconditional waiting time distribution .................... 60  

5 Expected waiting time in an M/G/1 priority queue .......... 62  
5.1 Waiting times .......................................................... 62  
5.2 Conservation laws ..................................................... 64
5.3 Time-dependent priorities ............................................. 67

6 The waiting time distributions in an accumulating priority M/G/1 queue
   6.1 Maximum priority process ........................................... 72
   6.2 Accreditation intervals ............................................. 75
   6.3 Priority distribution for an accredited customer .................. 78
   6.4 Waiting time distribution .......................................... 85

7 Numerical work ................................................................ 90

8 Conclusions ................................................................... 94
   8.1 Further work ............................................................ 95
Chapter 1

Introduction

Queueing is unavoidable in modern life. We queue to make purchases, to enter entertainment venues, on public transport and over the telephone.

Queueing models also have a wide variety of applications beyond these customer service environments. The field of queueing theory arose in telecommunications [1], and has since been applied in such diverse areas as job shop scheduling [2], modelling hard drive performance [3], and managing hospital emergency departments [4].

1.1 Queueing Models

Probabilistic models of queues are as diverse as their applications. In general, we assume that customers arrive according to some stochastic process. If the system is full at the time a customer arrives, then customers will queue until the system is able to serve them. Once a customer is in service, the time which the system takes to serve them is again random; after this, they depart from the system.

A helpful starting point is Kendall’s notation [5], which describes queues in the form $A/B/C$. The first component, $A$, describes the arrival process; customers are generally assumed to arrive according to some stochastic process, although a degenerate deterministic case is also allowed. Common arrival processes are Markovian or memoryless (M), deterministic (D), phase-type (PH), or general (G or GI — even a general arrival process is often assumed to have independent times between arrivals, as emphasised by the I). We restrict our attention to queues with memoryless (or Poisson) arrival processes.

The $B$ refers to the service distribution. Again, possible distributions are Markovian, deterministic, phase-type, general (only abbreviated G in this case), amongst other possibilities. Note that in all of these cases, service times are independent of each other, and of the arrival process. We consider general service time distributions in most of this thesis, occasionally restricting ourselves to Markovian (exponentially distributed) service times.

The $C$ in Kendall’s notation refers to the number of servers. If each server is
occupied, further arrivals will have to queue until a customer in service leaves. Typical server counts are 1, \( k \) (indicating that we want to solve the system for any number of servers), or \( \infty \) (in which case customers are always served immediately upon arrival).

A common extension to Kendall’s original notation takes the form \( A/B/C/K/M \) [6]. Here \( K \) is the number of customers who can be queued — arrivals beyond this limit will simply leave. Also, \( M \) is the total population of potential customers who may arrive to the system. Where \( K \) and \( M \) are unspecified, it is assumed that they are infinite, which will be the case throughout this thesis.

A further consideration is the service discipline. When multiple customers are queued and the server becomes idle, we need some rule for selecting the next customer to be served. Possible rules include first-come first-served, last-come first-served, selecting a queued customer at random, or selecting a customer based on their service time. Alternatively, the customers may be divided into different classes, which are treated differently by the service discipline. Such queues are called priority queues, and are the focus of this thesis.

When considering priority queues, we must also consider whether a queue is preemptive or not. In a preemptive queue, a customer currently in service can be removed if a higher priority customer arrives, while in a nonpreemptive queue any service underway must be completed before a new customer can be served [6].

Preemptive queues may be further classified into resuming and repeating types. In a preemptive resume queue, a customer ejected from service after time \( x_0 \) of a total service time \( x \) will return to service and require time \( x - x_0 \) to complete their service. The alternative is a preemptive repeat queue, where the customer must begin service again. These queues may further be divided into preemptive repeat without resampling, where the service time \( x \) is retained from the previous attempt at service, and preemptive repeat with resampling, where the service time is resampled from the service time distribution.

We will only consider non-preemptive queues, although we briefly discuss how the methods described in this thesis might apply to preemptive queues in the conclusion.

1.2 Overview

We begin in Chapter 2 by considering various background material in probability theory. We focus only on results which will be required in later chapters.

In Chapter 3 we consider a first-come, first-served M/G/1 queue, focusing on its analysis via busy periods as discussed by Kleinrock in [6]. This is the simplest M/G/1 priority discipline, but the methods used to determine its waiting time distribution will be reused on other priority disciplines in later chapters.

Next, we consider an M/G/1 queue with absolute priorities in Chapter 4, following the work of Conway, Maxwell and Miller in [2]. This is a system in which high priority customers are always selected for service over lower priority customers, regardless of how long the latter have been waiting.
In Chapter 5 we consider a method of determining the expected waiting time for a large class of M/G/1 priority queues, due to Kleinrock [7]. We focus in particular on accumulating priority queues, in which a customer’s priority is the product of their current waiting time and some constant determined by their priority class. This queue allows customers to overtake each other in priority.

Next, we consider very recent work by Stanford, Taylor and Ziedins [4] on accumulating priority queues in Chapter 6. While earlier work on this area was restricted to determining average waiting times, this work derives the Laplace-Stieltjes transform of the distribution.

We then demonstrate the utility of the techniques discussed in this thesis in Chapter 7, numerically deriving the probability distributions for an example accumulating priority queue. The methods used in this chapter apply with very little modification to first-come first-served and absolute priority cases as well.

Finally, in Chapter 8 we give our conclusions and a few suggestions for further investigation.
Chapter 2

Preliminaries

2.1 Laplace-Stieltjes Transforms

Laplace-Stieltjes transforms (LSTs) are an extension of the more familiar Laplace transform, and are closely related to other transforms used in probability, particularly the moment generating function. LSTs are particularly suited to dealing with distribution functions of nonnegative random variables, and we will make extensive use of them in the following chapters. This material is derived from Feller [8].

The Laplace-Stieltjes transform of a function $F$ is defined as

$$\tilde{F}(s) = \int_0^\infty e^{-sx} dF(x)$$

for all $s$ for which this integral converges.

We are primarily interested in the case where $F$ is the distribution function of a nonnegative random variable $X$, in which case it follows from Equation 2.1 that $\tilde{F}(s) = E(e^{-sx})$.

From this result, we can immediately observe that $\tilde{F}(0) = 1$ for any distribution function $F$. Moreover, if $Z = X + Y$, where $X$ and $Y$ are independent random variables with distribution functions $F_X$ and $F_Y$, then

$$\tilde{F}_Z(s) = \tilde{F}_X(s)\tilde{F}_Y(s)$$

for all $s$ for which both $\tilde{F}_X(s)$ and $\tilde{F}_Y(s)$ are defined.

If $F$ has a density $f$, then Equation 2.1 reduces to the Laplace transform

$$\tilde{F}(s) = \int_0^\infty e^{-sx} f(x) dx.$$  

It is common in queueing theory to encounter distributions which do not have a density; in particular waiting times generally have a point mass at zero, representing
the probability of an arrival encountering an empty system, while the remaining mass
is distributed across \((0, \infty)\). The use of the Riemann-Stieltjes integral in Equation 2.1
allows us to consider these cases. One could “differentiate” jumps in \(F\) to get Dirac
delta functions in \(f\) and recover some of the added flexibility of the Laplace-Stieltjes
transform without using Riemann-Stieltjes integrals, but we will not take this approach.

**Theorem 2.1.1.** If \(X\) is a nonnegative random variable with distribution function \(F\),
and \(\tilde{F}\) is \(n\) times differentiable at the origin, then

\[
E(X^n) = (-1)^n \tilde{F}^{(n)}(0). \tag{2.4}
\]

**Proof.** The integrand in Equation 2.1 is differentiable, bounded and continuous; thus
we can use differentiation under the integral to take the derivative of a Laplace-Stieltjes
transform.

\[
\tilde{F}'(s) = \int_0^\infty -xe^{-sx}dF(x) \tag{2.5}
\]

\[
\therefore \tilde{F}^{(n)}(s) = \int_0^\infty (-x)^n e^{-sx}dF(x). \tag{2.6}
\]

Substituting \(s = 0\) gives us expressions for any moment of \(X\).

\[
(-1)^n \tilde{F}^{(n)}(0) = \int_0^\infty x^n dF(x) \tag{2.7}
= E(X^n). \tag{2.8}
\]

If \(\tilde{F}\) does not have an \(n\)th derivative at the origin, then the corresponding moments
of \(X\) are undefined. We will generally require our probability distributions to have
Laplace-Stieltjes transforms in some neighbourhood of the origin.

Beyond the moments of \(X\), we may wish to determine its actual distribution function
from the LST. Feller derives an inversion formula in [8]:

\[
\sum_{k \leq sx} \frac{(-1)^k}{k!} s^k \tilde{F}^{(k)}(s) \to F(x) \tag{2.9}
\]

for any point \(x\) at which \(F\) is continuous.

From this, given a LST \(\tilde{F}\) we may numerically compute the original function \(F\)
to arbitrary precision. This procedure guarantees that no two distinct probability
distributions may have the same Laplace-Stieltjes transform. Most of our subsequent
results determine the Laplace-Stieltjes transforms of distribution functions, rather than
the distributions themselves; the existence of such inversion procedures makes such results meaningful.

Gaver [9] and Stehfest [10, 11] describe another method which depends only on the values of the transform itself, as opposed to its derivatives. Assume the original function $F$ is differentiable, with derivative $f$, and let $N$ be an even integer. The Gaver-Stehfest method uses $N$ points of the transform to approximate $f(t)$ by

$$f(t) \approx \frac{\log 2}{t} \sum_{i=1}^{N} V_i \tilde{F} \left( \frac{\log 2}{t^i} \right),$$

where

$$V_i = (-1)^{N/2+i} \sum_{k=\left\lfloor \frac{i+1}{2} \right\rfloor}^{\min(i,N/2)} \frac{k^{N/2+1}(2k)!}{(N/2-k)!k!(k-1)!(i-k)!(2k-i)!}. \tag{2.11}$$

Abate and Whitt [12] provide a third inversion method based on complex analysis, but we will not consider it here.

### 2.2 Wald’s Identity

Wald’s identity is a well-known result in probability theory. The derivation here is based on Wolff [13], but it can be found in any intermediate probability textbook.

**Theorem 2.2.1** (Wald’s Identity). Let $\{X_i|i \in \mathbb{N}\}$ be a sequence of independent random variables which share the same expectation $E(X_1) < \infty$. Let $N \geq 0$ be a random variable with the property that, for all $n \in \mathbb{N}$, the event $\{N > n\}$ is independent of $\{X_i|i > n\}$. Then

$$E \left( \sum_{i=1}^{N} X_i \right) = E(N)E(X_1). \tag{2.12}$$
Proof.

\[ E \left( \sum_{i=1}^{N} X_i \right) = E \left( \sum_{i=1}^{\infty} I_{\{N>n-1\}} X_n \right) \]  

(2.13)

\[ = \sum_{i=1}^{\infty} E \left[ E \left( I_{\{N>i-1\}} X_1 \ldots X_{i-1} \right) \right] \]  

(2.14)

\[ = \sum_{i=1}^{\infty} E \left( E \left( I_{\{N>i-1\}} \mid X_1 \ldots X_{i-1} \right) E \left( X_i \mid X_1 \ldots X_{i-1} \right) \right) \]  

(2.15)

\[ = \sum_{i=1}^{\infty} P(N > i - 1) E(X_i) \]  

(2.16)

\[ = E(X_1) \sum_{i=1}^{\infty} P(N > i - 1) \]  

(2.17)

\[ = E(X_1) E(N). \]  

(2.18)

Note that often in the statement of Wald’s identity, \( N \) is required to be a stopping time with respect to \( \{X_i \mid i \in \mathbb{N}\} \). A stopping time \( N \) for a sequence \( \{X_i : i \in \mathbb{N}\} \) satisfies \( \{N \leq n\} \in \sigma(X_1 \ldots X_n) \) for any integer \( n \); equivalently, if we know the values of \( X_1 \ldots X_n \), we can determine exactly whether \( N \leq n \). Our statement above applies when \( N \) is a stopping time, but also allows us to apply the theorem when somewhat weaker conditions hold, as we will see later.

For a trivial example, consider the case where \( N \) is independent of every \( X_i \); clearly \( N \) is not a stopping time for \( \{X_i : i \in \mathbb{N}\} \), but the conditions for this theorem are satisfied, and Equation (2.12) can be readily shown to hold.

### 2.3 Branching processes

Branching processes are a large subfield of probability, dealing with populations which change over time. We essentially only require one result, however, so our treatment will be very brief. An overview of branching processes can be found in many probability textbooks, for example [13] or [14]. This treatment is primarily based on [13].

Formally, a branching process is a sequence of random variables \( \{X_n : n \in \mathbb{Z}_{\geq 0}\} \), with \( X_0 = 1 \), along with a second family \( \{Y_{n,i} : n \in \mathbb{Z}_{\geq 0}, i \in \mathbb{N}\} \), which is IID and also independent of the first, satisfying

\[ X_{n+1} = \sum_{i=1}^{X_n} Y_{n,i}, \]  

(2.19)
Thus, every member $i$ of the population at time $n$ gives rise to some number $Y_{n,i}$ of individuals at time $n + 1$. Observe that, if $X_n = 0$ for any $n$, then $X_m = 0$ for all $m > n$.

**Theorem 2.3.1.** Let $X_n$ be a branching process, and $Y_{n,i}$ be the number of descendants of member $i$ in generation $n$. Denote the common expected number of descendants $E(Y_{n,i}) = \mu$, and let $\mu < 1$. Then:

1. $\lim_{n \to \infty} E(X_n) = 0$.
2. $X_n \xrightarrow{a.s.} 0$.
3. With probability 1, there is some $n$ such that $X_n = 0$.

**Proof.** We begin by calculating the expected population for any given time $n$.

$$
E(X_n) = E(E(X_n | X_{n-1})
$$

(2.20)

$$
= E(E(\sum_{i=1}^{X_{n-1}} Y_{n-1,i} | X_{n-1}))
$$

(2.21)

$$
= E(E(X_{n-1}Y_{1,1} | X_{n-1}))
$$

(2.22)

$$
= \mu E(X_{n-1}).
$$

(2.23)

Now $X_0 = 1$, so by induction for all nonnegative integers $n$ we have

$$
E(X_n) = \mu^n
$$

(2.24)

$$
\therefore \lim_{n \to \infty} E(X_n) = 0.
$$

(2.25)

This is the first result in the theorem above.

Markov’s inequality tells us that for any random variable $W$ and $w > 0$,

$$
P(|W| \geq w) \leq \frac{E(W)}{w}.
$$

(2.26)

We set $W = X_n$ and $w = 1$. Then

$$
P(X_n > 0) = P(X_n \geq 1)
$$

(2.27)

$$
\leq E(X_n)
$$

(2.28)

$$
= \mu^n.
$$

(2.29)

Thus, for $\mu < 1$, the probability of $X_n$ remaining nonzero decays exponentially. Let $A_n = \{X_n > 0\}$. Then $A_n$ is a decreasing sequence ($A_n \subset A_{n-1} \forall n \in \mathbb{N}$). Observe that $X_n$ remains positive for all $n$ if and only if $A_n$ occurs for all $n$. 
\[ P(\{X_n \neq 0\}) = P\left( \bigcap_{n=1}^{\infty} A_n \right) \]
\[ = \lim_{n \to \infty} P(A_n) \]
\[ \leq \lim_{n \to \infty} \mu^n \]
\[ = 0. \]

Thus, when the expected number of descendants \( \mu < 1 \), \( X_n \) converges almost surely to zero, proving the second part of the theorem. We note in passing that this result also holds if \( E(Y_{1,1}) = 1 \) and \( P(Y_{1,1} = 1) < 1 \) [14], but we will not make further use of this case.

Finally, note that the event \( \{X_n = 0\} \) is equivalent to \( A_n^c \), and let \( B \) be the event that there is some \( n \) for which \( X_n = 0 \). Then

\[ P(B) = P\left( \bigcup_{n=1}^{\infty} A_n^c \right) \]
\[ = 1 - P\left( \bigcap_{n=1}^{\infty} A_n \right) \]
\[ = 1. \]

This gives the third and final result in our theorem.

\[ \square \]

### 2.4 Renewal Theory

In this section we prove some of the basic results of renewal theory, which will be useful for analysing queues later on. Most of the material is adapted from Wolff [13], with some more technical material coming from Asmussen [15]. We have simplified some of Wolff’s material, particularly in considering only the identity function for reward processes where he allows arbitrary functions, and we regularly give more detailed derivations to assist the reader.

Let \( \{\tau_n : n \in \mathbb{N}\} \) be an independent and identically distributed family of positive random variables with common distribution function \( F \) and expectation \( \mu \), representing the duration of successive epochs, or intervals of time. We presume that epoch 0 starts at \( t = 0 \), and define the starting time of epoch \( n \in \mathbb{N} \) as

\[ T_n = \sum_{i=1}^{n} \tau_i. \]
Figure 2.1: Epoch durations, age and residual lifetime

From this we define a renewal process \( \{ N(t) : t \in \mathbb{R} \} \) to be the index of the epoch which is underway at time \( t \). Formally,

\[
N(t) = \max\{ n : t \geq T_n \}.
\]  

(2.38)

From this definition we can see that \( N(t) \) must always be a nonnegative integer, and as \( t \) increases \( N(t) \) can only ever increase or remain constant. If \( P(\tau_i = 0) = 0 \), then \( N(t) \) can only increase by 1 at any given time. Moreover, it follows from this definition that \( \{ N(t) \geq n \} = \{ T_n \leq t \} \). A renewal process can be viewed as a generalisation of a Poisson process, which we recover if the epoch durations are exponentially distributed.

Given some \( t \), the most recent jump was at time \( T_N(t) \), and the next jump will be at \( T_N(t+1) \). The time since the previous jump, or the age of this interval, is \( A(t) = t - T_N(t) \), while the residual lifetime of this interval is \( Y(t) = T_N(t+1) - t \). We show this graphically in Figure 2.1.

Applying the strong law of large numbers to Equation (2.37) shows that

\[
\frac{T_n}{n} \xrightarrow{a.s.} \mu.
\]  

(2.39)

Using this, we can show that \( N(t) \) is almost surely finite for any finite \( t \):

\[
P(N(t) = \infty) = \lim_{n \to \infty} P(N(t) \geq n)
\]  

(2.40)

\[
= \lim_{n \to \infty} P(T_n \leq t)
\]  

(2.41)

\[
= \lim_{n \to \infty} P \left( \frac{T_n}{n} \leq \frac{t}{n} \right)
\]  

(2.42)

\[
= 0.
\]  

(2.43)

However, if instead we fix \( n \) and let \( t \) increase without bound, we see that \( N(t) \xrightarrow{a.s.} \infty \).
\[ \lim_{t \to \infty} P(N(t) > n) = \lim_{t \to \infty} P(T_n < t) = \lim_{t \to \infty} P\left(\frac{T_n}{n} < \frac{t}{n}\right). \] 

(2.44)

Now assume for the purpose of a contradiction that the limit above is not 1. Then, taking \( t = 2n\mu \) shows that

\[ \lim_{n \to \infty} P\left(\frac{T_n}{n} \geq 2\mu\right) > 0. \]

(2.45)

This contradicts our earlier result that \( T_n/n \xrightarrow{a.s.} \mu \). Hence

\[ \lim_{t \to \infty} P(N(t) > n) = 1. \]

(2.46)

Finally, \( T_n \xrightarrow{a.s.} \infty \) as well:

\[ \lim_{n \to \infty} P(T_n > t) = \lim_{n \to \infty} P\left(\frac{T_n}{n} > \frac{t}{n}\right) = 1 \]

(2.47)

since \( T_n/n \xrightarrow{a.s.} \mu \).

#### 2.4.1 Limiting distribution of residual lifetime

We are interested in the behaviour of the process at arbitrary (but large) times. In particular, if we arrive to the system at some (possibly random) time \( t \), what is the distribution of \( Y(t) \), the remaining time in the current epoch?

We begin by defining \( C(t) \) as the amount of time up to \( t \) where \( Y(u) \leq y \), for some \( y > 0 \).

\[ C(t) = \int_0^t I_{\{Y(u) \leq y\}} du. \]

(2.50)

The reason for this definition of \( C(t) \) is that it gives us the distribution of residual lifetime for arrivals which are uniformly distributed over \([0, t]\), which we denote by \( F_0(y) \).
\[ E \left( \frac{C(t)}{t} \right) = \frac{1}{t} E \left( \int_0^t I_{Y(u) \leq y} \, du \right) \quad (2.51) \]

\[ = \frac{1}{t} \int_0^t E(I_{Y(u) \leq y}) \, du \quad (2.52) \]

\[ = \frac{1}{t} \int_0^t P(Y(u) \leq y) \, du. \quad (2.53) \]

We will find it useful to divide the period up to \( t \) into \( N(t) \) completed epochs, and the final partially completed epoch. During epoch \( i \), the interval in which \( Y(t) \leq y \) has length \( \min(y, \tau_i) \).

**Lemma 2.4.1.**

\[ E(\min(y, \tau_1)) = \int_0^y 1 - F(u) \, du. \quad (2.54) \]

**Proof.**

\[ E(\min(y, \tau_1)) = \int_0^\infty \min(x, y) \, dF(x) \quad (2.55) \]

\[ = \int_0^\infty \int_0^{\min(x,y)} 1 \, du \, dF(x). \quad (2.56) \]

We divide the limits of integration for \( x \) into two separate intervals, one where \( x \leq y \) and one where \( x > y \), avoiding the minimum term in the integral limits.

\[ E(\min(y, \tau_1)) = \int_0^y \int_u^x 1 \, du \, dF(x) + \int_y^\infty \int_0^y 1 \, du \, dF(x). \quad (2.57) \]

Next, we change the order of integration, which allows us to combine the two integrals and arrive at a final expression for our expectation.

\[ E(\min(y, \tau_1)) = \int_0^y \int_u^y 1 \, du \, dF(x) + \int_y^\infty \int_0^y 1 \, du \, dF(x) \quad (2.58) \]

\[ = \int_0^y \int_u^\infty 1 \, du \, dF(x) + \int_y^\infty \int_y^\infty 1 \, du \, dF(x) \quad (2.59) \]

\[ = \int_0^y 1 - F(u) \, du. \quad (2.60) \]

\[ \square \]

**Theorem 2.4.2.**

\[ \frac{C(t)}{t} \xrightarrow{a.s.} \frac{1}{\mu} \int_0^y 1 - F(>u) \, du. \quad (2.61) \]
Proof. We begin by considering the limit at renewal points as \( n \to \infty \); recall that as \( n \to \infty \), \( T_n \xrightarrow{a.s.} \infty \).

\[
C(T_n) = \sum_{i=1}^{n} \min(y, \tau_i) \tag{2.62}
\]

\[
C(T_n) = \frac{\sum_{i=1}^{n} \min(y, \tau_i) \ n}{T_n} \xrightarrow{a.s.} \frac{E(\min(y, \tau_1))}{\mu}. \tag{2.64}
\]

We have applied the Strong Law of Large Numbers to each fraction in Equation (2.63). As each fraction converges on a set of probability 1, the product of the two fractions will converge on (at least) the intersection of those sets, which will also have probability 1.

For general \( t \), \( C(t) \) has a more complicated form, because only part of the final interval is included. However, for large \( t \) the last interval is essentially negligible relative to all the previous intervals, and we can obtain a general form of the above convergence result.

\[
C(t) = \sum_{i=1}^{N(t)} \min(y, \tau_i) + \max(0, y - T_{n+1} - t). \tag{2.65}
\]

\[
\sum_{i=1}^{N(t)} \min(y, \tau_i) \leq C(t) \leq \sum_{i=1}^{N(t)+1} \min(y, \tau_i) \tag{2.66}
\]

\[
\frac{\sum_{i=1}^{N(t)} \min(y, \tau_i)}{T_{N(t)+1}} \leq \frac{C(t)}{t} \leq \frac{\sum_{i=1}^{N(t)+1} \min(y, \tau_i)}{T_{N(t)}}. \tag{2.67}
\]

We will show that both the lower and upper bound given above for \( C(t)/t \) above converge to the same limit. First, we consider the upper bound.

\[
\frac{\sum_{i=1}^{N(t)+1} \min(y, T_i)}{T_{N(t)}} = \frac{\sum_{i=1}^{N(t)} \min(y, T_i)}{T_{N(t)}} + \frac{\min(y, \tau_{N(t)+1})}{T_{N(t)}} \tag{2.68}
\]

\[
\leq \frac{\sum_{i=1}^{N(t)} \min(y, T_i)}{T_{N(t)}} + \frac{y}{T_{N(t)}} \tag{2.69}
\]

\[
\xrightarrow{a.s.} \frac{E(\min(y, \tau_1))}{\mu}. \tag{2.70}
\]

To see that the convergence above holds, recall that \( N(t) \xrightarrow{a.s.} \infty \). Thus, the strong law of large numbers applies to the first term in the same manner as in Equation (2.63), and the second converges to zero almost surely.
Now we consider the lower bound.

\[
\frac{\sum_{i=1}^{N(t)} \min(y, \tau_i)}{T_{N(t)+1}} = \frac{T_{N(t)}}{T_{N(t)+1}} \frac{\sum_{i=1}^{N(t)} \min(y, \tau_i)}{T_{N(t)}}
\]

\[
= \left(1 - \frac{T_{N(t)+1}}{T_{N(t)+1}}\right) \frac{\sum_{i=1}^{N(t)} \min(y, \tau_i)}{T_{N(t)}}
\]

\[
a.s. \rightarrow \frac{E(\min(y, \tau_1))}{\mu}.
\]  

(2.71)

In the first term, \(\tau_{N(t)+1}\) is finite with probability 1, while \(T_{N(t)+1} \rightarrow \infty\), so the entire term converges almost surely to 1. The second term is the same as in the upper bound.

Combining these results and substituting Equation (2.60), we have

\[
\frac{C(t)}{t} \rightarrow \frac{E(\min(y, \tau_1))}{\mu}
\]

\[
= \frac{1}{\mu} \int_0^y 1 - F(u) du.
\]

(2.74)

(2.75)

It turns out that the same convergence result holds in expectation as well. We will need another lemma (and Wald’s identity) to show this.

**Lemma 2.4.3.**

\[
\lim_{t \to \infty} \frac{E(N(t))}{t} = \frac{1}{\mu}.
\]

(2.76)

**Proof.** From the definition of \(T_n\) earlier,

\[
T_{N(t)+1} = \sum_{i=1}^{N(t)+1} \tau_i.
\]

(2.77)

Now for any \(t > 0\) and \(n \geq 0\), \(\{N(t) + 1 > n\}\) is independent of \(\{\tau_i : i > n\}\).

\[
\{N(t) + 1 > n\} = \{N(t) \geq n\} \quad (2.78)
\]

\[
= \{T_n \leq t\} \quad (2.79)
\]

\[
= \left\{\sum_{i=1}^{n} \tau_i \leq t\right\}. \quad (2.80)
\]
We can determine whether \( N(t) + 1 > n \) using only \( \tau_1, \ldots, \tau_n \), and future values of \( \tau_i \) are independent of these. Hence by Wald’s identity (Theorem 2.2.1),

\[
E(T_{N(t)+1}) = E(\tau_1)(E(N(t) + 1)) = \mu(E(N(t)) + 1). \tag{2.81}
\]

But from the definition of \( N(t) \) and \( Y(t) \), we have

\[
T_{N(t)+1} = t + Y(t). \tag{2.83}
\]

Taking expectations and equating with the previous expression:

\[
\mu E(N(t)) = t + E(Y(t)) - \mu \tag{2.84}
\]

\[
\frac{E(N(t))}{t} = \frac{1}{\mu} + \frac{E(Y(t))}{\mu t} - \frac{1}{t}. \tag{2.85}
\]

From here it may appear obvious that \( \frac{E(N(t))}{t} \to \frac{1}{\mu} \). However, we have not shown that \( E(Y(t)) \) is \( o(t) \), so some more work is required. We consider lower and upper bounds on the limit, in turn.

For the lower bound, note that \( Y(t) \) is always nonnegative. Thus, for any \( \epsilon > 0 \), we can choose some large \( T \) such that for all \( t > T \), we have

\[
\frac{E(N(t))}{t} \geq \frac{1}{\mu} - \epsilon \tag{2.86}
\]

\[
\therefore \liminf_{t \to \infty} \frac{E(N(t))}{t} \geq \frac{1}{\mu}. \tag{2.87}
\]

Now we consider the upper bound. Fix some \( c > 0 \) and let \( \tau^*_i = \min\{\tau_i, c\} \), \( \mu^* = E(\tau^*_1) \). We can construct a new renewal process from \( \{\tau^*_i | i \in \mathbb{N}\} \), which we denote by \( N^*(t) \). For any \( t > 0 \), \( N^*(t) \geq N(t) \), and hence also \( E(N^*(t)) \geq E(N(t)) \).

We can apply Equation (2.85) to this new process. We define the residual lifetime \( Y^*(t) \) as before, but for this process we always have \( Y^*(t) \leq c \).

\[
\frac{E(N^*(t))}{t} = \frac{1}{\mu^*} + \frac{E(Y^*(t))}{\mu^* t} - \frac{1}{t}. \tag{2.88}
\]

Thus the second and third terms above vanish for large \( t \), and

\[
\lim_{t \to \infty} \frac{E(N^*(t))}{t} = \frac{1}{\mu^*} \tag{2.89}
\]

\[
\limsup_{t \to \infty} \frac{E(N(t))}{t} \leq \frac{1}{\mu^*}. \tag{2.90}
\]
This holds for any \( c > 0 \); recall that \( \mu^* \leq \mu \). However as we take \( c \to \infty \), \( \mu^* \to \mu \), so for Equation (2.90) to hold, we must have

\[
\limsup_{t \to \infty} \frac{N(t)}{t} \leq \frac{1}{\mu}.
\]  

(2.91)

Combining Equations (2.87) and (2.91), we get

\[
\lim_{t \to \infty} \frac{N(t)}{t} = \frac{1}{\mu}.
\]  

(2.92)

**Theorem 2.4.4.**

\[
\lim_{t \to \infty} \frac{E(C(t))}{t} = \frac{1}{\mu} \int_0^y 1 - F(u) du.
\]  

(2.93)

**Proof.** The algebra is a little easier if we consider the renewal points immediately before and after \( t \).

\[
\sum_{i=1}^{N(t)} \min(y, \tau_i) \leq C(t) \leq \sum_{i=1}^{N(t)+1} \min(y, \tau_i)
\]  

(2.94)

\[-\min(y, \tau_{N(t)+1}) + \sum_{i=1}^{N(t)+1} \min(y, \tau_i) \leq C(t) \leq \sum_{i=1}^{N(t)+1} \min(y, \tau_i)
\]  

(2.95)

\[
-y + E\left(\frac{\sum_{i=1}^{N(t)+1} \min(y, \tau_i)}{t}\right) \leq \frac{E(C(t))}{t} \leq \frac{E\left(\sum_{i=1}^{N(t)+1} \min(y, \tau_i)\right)}{t}.
\]  

(2.96)

It is worth noting that while \( \{\tau_i : i \in \mathbb{N}\} \) are IID, \( \tau_{N(t)+1} \) does not share this distribution, because at any time \( t \) we are more likely to be in a long renewal epoch than a short one. However, our expression \( \min(y, \tau_{N(t)+1}) \) allows us to avoid worrying about the exact distribution of \( \tau_{N(t)+1} \) for now.

Now recall from the previous section that

\[
\{N(t) + 1 > n\} = \left\{ \sum_{i=1}^{n} \tau_i \leq t \right\}.
\]  

(2.97)

Thus, \( \{N(t) + 1 > n\} \) is independent of \( \{\min(y, \tau_i) : i > n\} \).\(^1\) Hence, by Wald’s identity,

\(^1\)Note that \( \{N(t) + 1 > n\} \) is not measurable with respect to \( \{\min(y, \tau_i) : i \leq n\} \), so it is not a stopping time for this sequence. This is why we used the more general form of Wald’s identity.
\[
E \left( \sum_{i=1}^{N(t)+1} \min(y, \tau_i) \right) = E(N(t) + 1)E(\min(y, \tau_1)). \tag{2.98}
\]

Substituting this expectation back into Equation (2.96) gives us
\[
-\frac{y}{t} + \frac{E(\min(y, \tau_1))}{t} + \frac{E(N(t))E(\min(y, \tau_1))}{t} \leq \frac{E(C(t))}{t} \tag{2.99}
\]
\[
\leq \frac{E(\min(y, \tau_1))}{t} + \frac{E(N(t))E(\min(y, \tau_1))}{t}. \tag{2.100}
\]

The upper and lower bounds above converge to the same limit as \( t \to \infty \), ensuring convergence:
\[
\lim_{t \to \infty} \frac{E(C(t))}{t} = \lim_{t \to \infty} \frac{E(N(t))E(\min(y, \tau_1))}{t}. \tag{2.101}
\]

Finally, we substitute in Equations (2.92) and (2.60).
\[
\lim_{t \to \infty} \frac{E(C(t))}{t} = \frac{1}{\mu} \int_0^\infty 1 - F(u) du. \tag{2.102}
\]

Together, Theorem 2.4.4 and Equation (2.53) give us the limiting distribution of residual lifetime for arrivals which are uniformly distributed over \([0, t]\), as \( t \to \infty \):
\[
\lim_{t \to \infty} P(Y_T \leq y) = \frac{1}{\mu} \int_0^y 1 - F(u) du, \quad T \overset{d}{=} U[0, t] \tag{2.103}
\]
\[
= F_0(y). \tag{2.104}
\]

This expression is a differentiable distribution function:
\[
\lim_{y \to \infty} F_0(y) = \frac{1}{\mu} \int_0^\infty 1 - F(u) du = \frac{E(\tau_1)}{\mu} = 1. \tag{2.105}
\]
\[
dF_0(y) = \frac{1 - F(y)}{\mu}, \quad y > 0. \tag{2.106}
\]
\[
\frac{dF_0(y)}{dy} = \frac{1 - F(y)}{\mu}, \quad y > 0. \tag{2.107}
\]
In fact, Equation (2.103) also gives the limiting distribution of $Y(t)$ for any $t$ (not necessarily uniformly distributed), as long as the epoch durations have non-lattice distributions. The proof of this result is highly technical. An analytical version is given by Feller in [8]; Asmussen gives a probabilistic proof in [15], as well as a similar derivation to Feller’s.

If the epoch durations are lattice, $Y(t)$ could have a periodic form for all $t$. For example, if $\tau_i = 1$ with probability 1, then $Y(t) = 1$ for all $t \in \mathbb{N}$. However, the periodicity will be smoothed out for $Y_T$ at uniformly distributed $T$, so Equation (2.103) still holds.

We will now assume that the epoch durations are non-lattice. Our focus in later chapters will be queues with Poisson arrivals and service times do not depend on the arrival process, so we will not encounter any lattice-type output processes from such queues.

Denoting the limiting distribution by $F_Y$, we have

$$F_Y(y) = \lim_{t \to \infty} P(Y(t) \leq y) = \frac{1}{\mu} \int_0^y 1 - F(u) du. \quad (2.108)$$

We can also find the LST of $F_Y$.

$$\tilde{F}_Y(s) = \frac{1}{\mu} \int_0^\infty (1 - F(y))e^{-sy} dy \quad (2.109)$$
$$= \frac{1}{\mu} \left[ (1 - F(y))e^{-sy}\right]_0^\infty + \frac{1}{s} \int_0^\infty f(y)e^{-sy} dy \quad (2.110)$$
$$= \frac{1 - \tilde{F}(s)}{\mu s}. \quad (2.111)$$

We may use the LST to find the expectation of the limiting distribution $Y$.

$$E(Y) = -\lim_{s \to 0} \frac{d}{ds} \tilde{F}_Y(s) \quad (2.112)$$
$$= \lim_{s \to 0} \tilde{F}(s) - s\tilde{F}'(s) - 1 \quad (2.113)$$
$$= \lim_{s \to 0} \frac{s\tilde{F}''(s)}{2\mu s} \quad (2.114)$$
$$= \frac{E(\tau^2)}{2E(\tau)}. \quad (2.115)$$

Note that we have not demonstrated that this is the same as the limiting expectation of $Y_t$. However, Asmussen [15] shows that this is in fact the case:

$$\lim_{t \to \infty} E(Y_t) = \frac{E(\tau^2)}{2E(\tau)}. \quad (2.116)$$
2.4.2 Joint distributions

Observe that \( \{ Y(t) > y, A(t) > z \} = \{ Y_{t-z} > y + z \} \). From here, we can determine the limiting joint distribution of \( Y(t) \) and \( A(t) \).

\[
\lim_{t \to \infty} P(Y(t) > y, A(t) > z) = \lim_{t \to \infty} P(Y_{t-z} > y + z) = \frac{1}{\mu} \int_{y+z}^{\infty} 1 - F(u) du.
\]

(2.117)

(2.118)

We can also determine the limiting distribution of \( A(t) \) by substituting \( y = 0 \) into this expression. Note that \( A(t) \) and \( Y(t) \) have the same limiting distribution; intuitively, if we pick a large time it is equally likely to be near the start or the end of an epoch.

\[
\lim_{t \to \infty} P(A(t) > z) = \frac{1}{\mu} \int_{z}^{\infty} 1 - F(u) du.
\]

(2.119)

We have the limiting joint distribution of \( A(t) \) and \( Y(t) \) above, but where possible we would also like their limiting joint density, which we will denote by \( f_{A,Y} \). Using the law of total probability twice we write

\[
P(A(t) \leq z, Y(t) \leq y) + P(A(t) \leq z, Y(t) > y) = P(A(t) \leq z)
\]

(2.120)

\[
P(A(t) > z, Y(t) > y) + P(A(t) \leq z, Y(t) > y) = P(Y(t) > y).
\]

(2.121)

Subtracting the second line from the first gives

\[
P(A(t) \leq z, Y(t) \leq y) - P(A(t) > z, Y(t) > y) = P(A(t) \leq z) - P(Y(t) > y).
\]

(2.122)

Observe that each term on the right depends on only one of \( z \) and \( y \). Thus, assuming the distribution of renewal times \( F \) is differentiable, with derivative \( f \):

\[
f_{A,Y}(x,y) = \frac{\partial^2}{\partial x \partial y} \lim_{t \to \infty} P(A(t) \leq x, Y(t) \leq y) = \frac{\partial^2}{\partial x \partial y} \lim_{t \to \infty} P(A(t) > x, Y(t) > y) = \frac{1}{\mu} \frac{\partial}{\partial y} (F(x+y) - 1) = f(x+y) \frac{1}{\mu}, \ x, y \geq 0.
\]

(2.123)

(2.124)

(2.125)

(2.126)
Given fact that \( \tau_{N(t)} = A(t) + Y(t) \), we can also determine \( f_{\tau,Y} \), the limiting joint density of \( \tau_{N(t)} \) and \( Y(t) \), with a simple change of variables.

\[
f_{\tau,Y}(z,y) = \frac{f(z)}{\mu}, \quad 0 \leq y \leq z. \tag{2.127}
\]

As a final observation, we can determine the limiting distribution of \( \tau_{N(t)} \). Note that this is distinct to the distribution of \( \tau_1 \). The reason for this is that, at any given time, we are relatively more likely to be in a long epoch than a short one.

\[
\lim_{t \to \infty} f_{\tau_{N(t)}}(z) = \int_0^z f_{\tau,Y}(z,y)dy = \frac{zf(z)}{\mu}, \quad z \geq 0. \tag{2.128}
\]

\[
\lim_{t \to \infty} \frac{1}{t} \int_0^t N(u)du. \tag{2.130}
\]

We denote the limit of this value by \( L \), where it exists:

\[
L = \lim_{t \to \infty} \frac{1}{t} \int_0^t N(u)du. \tag{2.131}
\]
The following theorem is essentially a more verbose version of Wolff’s in [13].

**Theorem 2.5.1.** Assume that there are positive real numbers $\lambda$ and $w$ such that

\[
\frac{\Lambda(t)}{t} \xrightarrow{a.s.} \lambda \quad (2.132)
\]

\[
\frac{1}{n} \sum_{j=1}^{n} W_j \xrightarrow{a.s.} w. \quad (2.133)
\]

Then the limit in Equation (2.131) exists almost surely, and the limiting average number of customers in the system over time is

\[
L = \lambda w. \quad (2.134)
\]

**Proof.** We begin by establishing bounds on the total time spent in the system up to time $t$ by considering waiting times. Lower and upper bounds are given by the total waiting time of every customer who either leaves before $t$, or arrives before $t$, respectively. (If a customer $j$ is still in the system at time $t$, they will have waited for some period between 0 and $W_j$, making both of these bounds inexact.) Dividing by $t$ to recover time averages, we have

\[
\frac{1}{t} \sum_{j: \tau_j + W_j \leq t} W_j \leq \frac{1}{t} \int_{0}^{t} N(u)ds \leq \frac{1}{t} \sum_{j=1}^{\Lambda(t)} W_j. \quad (2.135)
\]

The upper bound is readily evaluated:

\[
\frac{1}{t} \sum_{j=1}^{\Lambda(t)} W_j = \frac{\Lambda(t)}{t} \frac{1}{\Lambda(t)} \sum_{j=1}^{\Lambda(t)} W_j. \quad (2.136)
\]

By Equation (2.132), $\Lambda(t) \xrightarrow{a.s.} \infty$. Hence each term in the product above converges on a set of probability 1. The intersection of these sets also has probability 1, and thus

\[
\frac{1}{t} \sum_{j=1}^{\Lambda(t)} W_j \xrightarrow{a.s.} \lambda w. \quad (2.137)
\]

The lower bound is a little more involved. By Equations (2.132) and (2.133), respectively,
\[ W_n \xrightarrow{\text{a.s.}} 0. \quad (2.139) \]

Using these results, and noting that \( \Lambda(t_n) = n \),

\[
\frac{W_n}{\tau_n} = \frac{W_n n}{n \tau_n} - \frac{W_n \Lambda(t_n)}{n \tau_n} \xrightarrow{\text{a.s.}} 0. \tag{2.142}
\]

Thus, for any \( \omega \in \Omega \) and \( \epsilon > 0 \), there is some \( m \) such that, for all \( j > m \), \( W_j(\omega) < \tau_j(\omega)\epsilon \), or equivalently \( \tau_j(\omega) + W_j(\omega) < \tau_j(\omega)(1 + \epsilon) \).

Let \( J = \{ j : \tau_j(\omega) + W_j(\omega) \leq t \} \), the index of each arrival up to \( t \), as in the lower bound in Equation (2.135). Then, consider \( J' = \{ j : j > m, \tau_j(\omega)(1 + \epsilon) \leq t \} \). Note that \( J' \subset J \); the first \( m \) arrivals have been removed, and the definition of \( \epsilon \) above means that we stop counting arrivals at some instant before \( t \).

\[
\frac{1}{t} \sum_{j \in J} W_j(\omega) \geq \frac{1}{t} \sum_{j \in J'} W_j(\omega) \geq \frac{1}{t} \sum_{j=1}^{\Lambda(t')/t} W_j(\omega) - \frac{1}{t} \sum_{j=1}^{m} W_j(\omega) \tag{2.144}
\]

where \( t' = \frac{t}{1+\epsilon} \). The first term above is simply a scaled version of the upper bound in Equation (2.135). The second term converges to zero for almost all \( \omega \in \Omega \) as, with probability 1, waiting times are finite while the arrival count diverges. Thus, for almost all \( \omega \),

\[
\lim \inf_{t \to \infty} \frac{1}{t} \sum_{j \in J} W_j(\omega) \geq \frac{\lambda w}{1 + \epsilon}. \tag{2.146}
\]

This holds for any \( \epsilon > 0 \), so for almost all \( \omega \)

\[
\lim \inf_{t \to \infty} \frac{1}{t} \sum_{j \in J} W_j(\omega) \geq \lambda w. \tag{2.147}
\]
This establishes that the limiting lower bound for Equation (2.135) is also $\lambda w$. Hence, the average arrival rate converges almost surely to the same value.

As well as the obvious interpretation in terms of queue length, Little’s law can be applied to describe the proportion of time which a queueing system spends in a given state.

**Theorem 2.5.2.** Consider a stable single-server queue, where customers are divided into one of $P$ classes. If customers of class $i$ form a renewal process with mean inter-arrival time $1/\lambda_i$, and class $i$ service durations have mean $1/\mu_i$, the proportion of time which the server is busy with class $i$ customers is

$$\rho_i = \frac{\lambda_i}{\mu_i}. \quad (2.148)$$

**Proof.** We use Little’s law, but consider only class $i$ customers who are currently in service.

For a stable system, the long-run rate at which class $i$ customers arrive to the server is the same as the rate at which they arrive to the system as a whole. The time a class $i$ customer spends in this system is simply the mean service duration. The number $N_i$ of class $i$ customers in this system can only ever be zero or one, depending on whether the server is serving a class $i$ customer or not. Finally, the limiting average waiting time and arrival count will converge almost surely by the Strong Law of Large Numbers; the “waiting time” here is the time the server takes to serve a customer. Note that the renewal process includes Poisson arrivals, as well as other cases where inter-arrival times are independent.

Thus $E(N_i) = \rho_i$, where $\rho_i$ is the proportion of time which the system spends on class $i$ customers. The result then follows from Theorem 2.5.1, with arrival rate $\lambda_i$ and mean time in system $1/\mu_i$.

Note that by setting $P = 1$ above we see that the proportion of time that a single-server system is busy is equal to $\rho = \lambda/\mu$, which we call the *utilisation* of the queue.

It is not entirely straightforward to rigorously demonstrate that the conditions for Little’s law hold; for example, Kleinrock [6] largely avoids the question of whether the mean service times converge. Independent arrivals form a renewal process, so the limit $L$ exists by Lemma 2.4.3. However, it is not easy to rigorously demonstrate that the mean waiting time converges to some limit $W$, although it is perhaps intuitively clear. We omit the full proof here, but note that Asmussen gives a rigorous derivation of Little’s law for the GI/G/1 queue (and hence also in the M/G/1 case which we are considering) in [15].

For the purposes of outlining a proof, we note that Theorem 2.5.1 holds in a more general sense; if any two of the limits $L$, $\lambda$ and $W$ can be shown to exist (and are nonzero) then the third must also exist, and $L = \lambda W$. 

25
Asmussen shows in [15] that the limit $L$ exists by regenerative arguments similar to those in Section 2.4. In an M/G/1 queue, whenever a busy period ends, system is effectively reset, and the future evolution of the system is independent of any past behaviour. In the GI/G/1 queue this is not the case, but weaker regenerative conditions still hold, so for nonlattice arrivals the queue length has a single limiting stationary distribution. From there it can be shown that $L$ exists, and is equal to $\lambda W$. We omit the full proof, and refer to [15] for details.
Chapter 3

First-come, first-served M/G/1 queue

The first queue we will consider is the M/G/1 first-come, first-served (FCFS) queue. Our analysis is based primarily on Kleinrock’s busy period analysis in [6], which in turn draws upon Gaver’s busy period analysis in [17] and Takács’s LCFS approach to determining the busy period length in [18]. There are several other methods of analysing the M/G/1 FCFS queue; Kleinrock primarily covers the imbedded Markov chain method in [6]. We have chosen the busy period method because it is useful in analysing M/G/1 queues with more complicated priority disciplines, as we will see in the following chapters.

In this model, customers arrive, one at a time, at random times. We denote the arrival time of customer \( n \) by \( T_n \). The inter-arrival times \( \tau_n \) are defined by

\[
\tau_1 = T_1; \\
\tau_n = T_n - T_{n-1}, \quad n > 1.
\]

(3.1) (3.2)

The “M” in the queue description specifies that the inter-arrival times have independent memoryless, or exponential, distributions with parameter \( \lambda \). Equivalently, the number of arrivals up to time \( t \) forms a Poisson process with parameter \( \lambda \).

The server can process a single customer at a time; hence the “1” in the queue description. If a customer arrives when the server is empty, they will immediately enter service. Once they have completed service, the customer exits the system. If a customer arrives when another customer is in service, they must wait until the server finishes; this queue is non-preemptive. There are pre-emptive queueing models where a customer can displace another from service, but we will not consider such systems here.

When a customer finishes service and there are multiple candidates for the next customer to be processed, the server must decide who to serve next. This process is known as the queueing discipline. Here we are considering a first-come first-served queue, so the server will simply choose the customer who has been waiting the longest.
Later, we will consider priority queues, where customers are assigned priorities, which can be independent of their order of arrival. The time between the arrival time \( T_n \), and the time that customer \( n \) enters service is known as the waiting time, which we denote by \( W_n \). Clearly the waiting time is random, depending on the number of customers who are waiting in the queue at \( T_n \), and on their service times. Moreover, it has a mixed distribution - it will be zero with some positive probability representing the possibility that the queue is idle at \( T_n \), and otherwise will have a distribution over the positive real numbers.

The service time of customer \( n \) is denoted by \( X_n \). Service times are independent and identically distributed with some distribution function \( B(x) \) and mean \( \mu^{-1} \). We require that \( B \) has a Laplace-Stieltjes transform \( \tilde{B}(s) \) in some neighbourhood of the origin. The unspecified, or general, service time distribution explains the “G” in the queue description.

We give an example realisation of an M/G/1 FCFS queue in Figure 3.1. The queue is initially empty. At \( t = 3 \) customer \( C_1 \) arrives, and immediately enters service, as the server is idle. This customer has service time \( X_1 = 2 \), so \( C_1 \) remains in service until \( t = 5 \). No customers arrive during this period, so the server is idle from \( t = 5 \) until \( C_2 \) arrives at \( t = 6 \). While \( C_2 \) is in service, customers \( C_3 \) and \( C_4 \) arrive at \( t = 7 \) and \( t = 8 \), respectively. These customers have to wait in the queue before they are served. Customer \( C_2 \) has waiting time \( X_2 = 4 \), so \( C_3 \) must wait until \( t = 10 \) to be served, meaning that the waiting time \( W_3 = 3 \). The last customer \( C_4 \) is served after \( C_3 \) finishes service at \( t = 13 \), so the waiting time \( W_4 = 5 \). No further customers arrive before \( C_4 \) finishes service, so the system becomes idle again at \( t = 15 \).

### 3.1 Derived processes

The processes \( \{N(t)\}_{t \geq 0} \), the number of arrivals up to time \( t \), and \( \{X_n\}_{n \in \mathbb{N}} \), the service time of each customer, are “load” processes — inputs to the queueing system. We are interested in “output” processes, which are derived from the load processes. Examples of output processes are waiting times and queue lengths. However, one can gain considerable insight into this and other queues by examining more elaborate stochastic processes.

#### 3.1.1 Virtual Work

We define the virtual work \( U(t) \) to be the time required after time \( t \) to complete the service of every customer currently in the system at \( t \) (ignoring any future arrivals). In this system, an arrival at time \( t \) will enter service exactly when all customers presently in the system have completed service, so \( U(t) \) can be considered to be the waiting time for an arrival at time \( t \). This property does not hold for other queueing disciplines, although as we shall see, the virtual work process is relevant in these cases as well.

Note that the system is busy at time \( t \) if and only if \( U(t) > 0 \). Whenever the system
Figure 3.1: Sample operation of an M/G/1 FCFS queue. Note that $N_A(t)$ is number of arrivals up to time $t$, and $N_S(t)$ is the number in the system at time $t$. 
is busy, it will be serving a customer, and so the virtual work will decrease at a uniform rate of 1. The only exception is at arrival times $T_n$, when the work will jump by the service time $X_n$. The virtual work will never be negative; when it reaches zero the system is idle, and it will remain at zero until a new customer arrives.

The final observation we shall make about the virtual work is that it is independent of the queueing discipline, as long as the queue is work-conserving. It decreases at rate 1 whenever the system is busy, and increases whenever customers arrive; this occurs regardless of which customer is selected for service. Thus, when the system is busy the actual customers whose service times make up the virtual work may differ depending on the queueing discipline, but the amount of work required to serve them will not.

We give an example of the virtual work process for the queue described earlier, in Figure 3.1. Note that the virtual work jumps at times 3, 5, 6 and 8, corresponding to the arrival of new customers. The value of the jumps are the service time of the newly arrived customers. Also observe that customer departures do not affect the virtual work process, unless the departure leaves the system idle.

It is possible to derive the LST of the stationary waiting time using the virtual work process. We cover this briefly below. However, this analysis does not readily generalise to other queueing disciplines, so we will devote more attention to the busy period analysis later in this chapter.

Given a Markov process $\{X_t\}_{t \geq 0}$, we define an operator $A$, known as the infinitesimal generator [8]. Its operation on a function $h$ is given by

$$A h(x) = \lim_{\delta \to 0} \frac{1}{\delta} (E(h(X_{t+\delta})|X_t = x) - h(x)).$$

(3.3)

For all functions $h$ in the domain of $A$, a stationary distribution $X_\infty$ satisfies

$$E(A h(X_\infty)) = 0.$$  

(3.4)

Different authors specify different restrictions on the function $h$ [15], but we are looking for the Laplace-Stieltjes transform of $V_\infty$, so we will only require the case where $h(v) = e^{-sv}$. This $h$ is bounded and uniformly continuous for nonnegative $v$ (and our process can never be negative), allowing us to avoid more technical considerations regarding the domain of $A$.

Consider the virtual work process $\{V(t)\}_{t \geq 0}$. We need to consider two cases: if $V(t) > 0$, the process uniformly decreases at rate 1, while if $V(t) = 0$, the process has no drift. In either case, $V$ will increase by a random value $X$ (a single service time) if there is an arrival. We are concerned with a small interval $(t, t + \delta)$, so the probability of more than one arrival is vanishingly small.

For strictly positive values of $v$: 

30
\[ A h(v) = \lim_{\delta \to 0} \frac{1}{\delta} (h(v - \delta)(1 - \lambda \delta + o(\delta)) + E(h(v + X - \delta))(\lambda \delta + o(\delta)) - h(v)) \quad (3.5) \]

\[ = \lim_{\delta \to 0} \frac{h(v - \delta) - h(v)}{\delta} - \lambda h(v) + \lambda E(h(X + v)) \quad (3.6) \]

\[ = \lambda E(h(X + v)) - \lambda h(v) - h'(v), \quad v > 0. \quad (3.7) \]

When \( v = 0 \):

\[ A h(0) = \lim_{\delta \to 0} \frac{1}{\delta} (h(0)(1 - \lambda \delta + o(\delta)) + E(h(X))(\lambda \delta + o(\delta)) - h(0)) \quad (3.8) \]

\[ = \lambda E(h(X)) - \lambda h(0). \quad (3.9) \]

It is more convenient to express this in a similar form to the case for positive \( v \), however.

\[ A h(v) = \lambda E(h(X + v)) - \lambda h(v) - h'(v) + h'(0), \quad v = 0. \quad (3.10) \]

We can now take the expectation of the generator at our stationary distribution

\[ E(\mathbb{A} h(V_\infty)) = E(\mathbb{A} h(V_\infty)I(V_\infty = 0)) + E(\mathbb{A} h(V_\infty)I(V_\infty > 0)) \quad (3.11) \]

\[ = E(\lambda h(X + V_\infty) - \lambda h(V_\infty) - h'(V_\infty)) + E(h'(0)I(V_\infty = 0)) \quad (3.12) \]

\[ = E(\lambda h(X + V_\infty) - \lambda h(V_\infty) - h'(V_\infty)) + h'(0)P(V_\infty = 0). \quad (3.13) \]

Now \( P(V_\infty = 0) = 1 - \rho \), and we are only considering the case where \( h(v) = e^{-sv} \), so

\[ E(\mathbb{A} h(V_\infty)) = E(\lambda e^{-s(X + V_\infty)} - \lambda e^{-sV_\infty} + se^{-sV_\infty}) - s(1 - \rho) \quad (3.14) \]

\[ = E(e^{-sV_\infty})(\lambda E(e^{-sX}) - \lambda + s) - s(1 - \rho). \quad (3.15) \]

In the second equality we have used the independence of \( X \) and \( V_\infty \); here \( X \) represents an arrival to a system which is already in the stationary distribution \( V_\infty \).

Now recall that \( \bar{B}(s) = E(e^{-sX}) \), and that for a stationary distribution, \( E(\mathbb{A} h(V_\infty)) = 0 \). Thus we can solve the last equation to determine the LST of the stationary distribution of virtual work in terms of the LST of service time.

\[ E(e^{-sV_\infty}) = \frac{s(1 - \rho)}{\lambda \bar{B}(s) - \lambda + s}. \quad (3.16) \]

For a FCFS queue, the virtual work is the same as the waiting time. This gives us the LST of waiting time, which we will later derive using the busy period in Theorem 3.3.3.
### 3.1.2 Maximum waiting time

The maximum waiting time process is the maximum possible time that a customer (not currently in service) could have been waiting, given that we can observe the server, but not the queue. That is, we know the time of arrival and service commencement for all customers who are being or have been served, but have no information about customers who are currently queued.

If the server is empty, then the maximum waiting time must be zero; our system will always serve a customer if one is available. If the time is currently \( t \) and the server is occupied by a customer who arrived at time \( s \), there could be some other customer who arrived at any time after \( s \), so the maximum possible waiting time is \( t - s \). Thus, at the time a customer \( C \) goes into service, the maximum waiting time process is the length of time since \( C \) arrived, which is \( C \)'s waiting time. While a customer is in service the maximum waiting time process will increase at a constant rate of 1.

We will not make further use of this process in this chapter. However, a generalisation of the maximum waiting process known as the maximum priority process is central to the analysis of the accumulating priority queue in Chapter 6.

### 3.2 Busy and idle periods

The operation of a queue can be characterised by alternating idle and busy periods. We begin our analysis of the M/G/1 FCFS queue by determining the distribution of the length of these periods.

**Theorem 3.2.1.** The DF of an idle period \( F(y) \) in an M/G/1 queue satisfies

\[
F(y) = 1 - e^{-\lambda y}.
\]

_Proof._ The length of idle periods are IID, by the memoryless arrival property. Once the system is idle, it will remain so until a new customer arrives. The duration from any time \( t \) until the next customer arrives is exponential with parameter \( \lambda \), giving \( F(y) \) above.

Busy period lengths are also IID. A busy period commences when a customer arrives to an empty system. From there, the length of the busy period depends only on future arrivals during that busy period, which form a Poisson process, and the service times of future customers, which are IID with DF \( B(x) \). The length of any given busy period depends only on the arrivals to that particular busy period.

Actually determining the distribution of a busy period duration is more involved. Recall that we can determine from the virtual work whether the queue is busy or idle, and that the virtual work process does not depend on the queueing discipline. Therefore, while we are interested in a FCFS queue, we can choose a different queueing
Consider Figure 3.2. At $t = 0$, customer $C_0$ arrives to an empty system. While $C_0$ is in service, $C_1$ and $C_2$ arrive at $t = 1$ and $t = 4$, respectively. At $t = 7$, $C_0$ finishes service, and the most recently arrived customer, $C_2$, enters service. While $C_2$ is in service, $C_3$ arrives at $t = 9$; then $C_2$ concludes service at $t = 10$ and again the last customer to arrive, $C_3$, enters service. Finally $C_3$ finishes service at $t = 12$ and the most recently arrived customer is $C_1$, who enters service and finishes at $t = 14$. At this point there are no more customers queued, so the system becomes idle again.

The important property of the LCFS system is that every new customer who enters service generates a sub-busy period with the same distribution as an entire busy period. In the example above, $C_1$ was delayed not only by $C_2$, but also by $C_3$, who arrived while $C_2$ was in service. If more customers had arrived while $C_3$ was in service, $C_1$ would have had to wait for them, and for any customers who arrived while they were in service, and so on. Such a sequence of customers has exactly the same distribution as an entire busy period.

Theorem 3.2.2. The distribution of busy period length in an M/G/1 queue satisfies

$$\bar{G}(s) = \bar{B}(s + \lambda - \lambda \bar{G}(s)).$$  

(3.18)

Proof. Denote the busy period length by $T_b$, the service time of the first customer in the busy period by $X_0$, and the number of new arrivals while the first customer is in service by $N$. Each of these arrivals generates their own sub-busy period with the same distribution as the original busy period. The duration of the busy period generated by the $i^{th}$ interval is given by $Y_i$. 

Figure 3.2: Division of the busy period of an M/G/1 FCFS queue
\[ T_b = X_0 + \sum_{i=1}^{N} Y_i. \]  

(3.19)

The \( Y_i \) and \( N \) are independent; \( N \) depends on the number of arrivals in the first interval, while each \( Y_i \) depends on a different set of service times, all of which occur after the first interval.

We take the Laplace-Stieltjes transform of this expression, and condition on \( X_0 \) and \( N \).

\[
E(\exp(-sT_b | X_0 = x, N = k)) = E\left[\exp\left\{-s(x + \sum_{i=1}^{k} Y_i)\right\}\right] 
= E(\exp(-sx \prod_{i=1}^{k} e^{-sY_i})) 
= E(\exp(-sx) \prod_{i=1}^{k} E(\exp(-sY_i))) 
= e^{-sx}[\tilde{G}(s)]^k. 
\]

(3.20)

(3.21)

(3.22)

(3.23)

Now \( N \) is the number of arrivals to the system while the original customer is in service, which is an interval of length \( x \) by our conditioning. For a queue with memoryless arrivals, the number of arrivals in a period of length \( x \) has a Poisson distribution with parameter \( \lambda x \); recall that the customer arrival process is independent of all service times.

\[
E(\exp(-sT_b | X_0 = x)) = \sum_{k=0}^{\infty} E(\exp(-sT_b | X_0 = x, N = k))P(N = k) 
= \sum_{k=0}^{\infty} e^{-sx}[\tilde{G}(s)]^k \frac{(\lambda x)^k}{k!}e^{-\lambda x} 
= \exp(-x[s + \lambda - \lambda\tilde{G}(s)]). 
\]

(3.24)

(3.25)

(3.26)

In the above expression \( X_0 \) is simply the service time of the first customer, which has distribution \( B(x) \). We can use this to remove the last condition on the expectation above, and obtain our desired functional equation for \( \tilde{G}(s) \).

\[
E(\exp(-sT_b)) = \int_0^{\infty} E(\exp(-sT_b | X_0 = x))dB(x) 
= \int_0^{\infty} \exp(-x[s + \lambda - \lambda\tilde{G}(s)])dB(x) 
\tilde{G}(s) = \tilde{B}(s + \lambda - \lambda\tilde{G}(s)). 
\]

(3.27)

(3.28)

(3.29)
From Equation (3.29), we may approximate \( \tilde{G}(s) \) for any \( s \). Wolff [13] shows that, for any fixed \( s \), we can obtain a sequence \( \tilde{G}_n(s) \rightarrow \tilde{G}(s) \) by iteratively applying the formula

\[
\tilde{G}_{n+1}(s) = \tilde{B}(s + \lambda - \lambda \tilde{G}_n(s)). \tag{3.30}
\]

Given a numerical approximation to \( \tilde{G}(s) \), we may then use the inversion methods of Section 2.1 to numerically find the distribution function \( G \).

**Corollary 3.2.3.** The mean length of a busy period satisfies

\[
\frac{1}{\mu_b} = \frac{1}{\mu(1 - \rho)}. \tag{3.31}
\]

**Proof.** By Theorem 2.1.1, we may find the mean busy period length by differentiating its LST. Recall that \( \rho = \frac{\lambda}{\mu} \).

\[
\frac{1}{\mu_b} = -\frac{d\tilde{G}(s)}{ds} \bigg|_{s=0}
\]

\[
= (1 - \lambda \tilde{G}'(0))\tilde{B}'(\lambda - \lambda \tilde{G}(0))
\]

\[
= \frac{1 + \lambda/\mu_b}{\mu}
\]

\[
\therefore \frac{1 - \rho}{\mu_b} = \frac{1}{\mu}
\]

\[
\frac{1}{\mu_b} = \frac{1}{\mu(1 - \rho)}. \tag{3.36}
\]

\[ \Box \]

### 3.3 Waiting times

We now consider the distribution of waiting times, which vary with the service discipline, so we will return to the FCFS queue.

This analysis proceeds by splitting the busy period into sub-periods. Let period 0 be the interval in a busy period when the first customer is served, and call its duration \( S_0 \). Then let period \( i \) be the interval in which all arrivals from period \( i - 1 \) are served, with duration \( S_i \).

We show an example of this in Figure 3.3. At time 0, customer \( C_0 \) arrives to an empty system. This customer immediately enters service, and its service interval is
Figure 3.3: Determining the waiting time distribution of an M/G/1 FCFS queue

period 0, lasting until \( t = 6 \). Two customers, \( C_1 \) and \( C_2 \), arrive during this period, at \( t = 3 \) and \( t = 4 \) respectively. At \( t = 6 \), customer \( C_0 \) leaves service, and the second period begins. Because \( C_1 \) and \( C_2 \) arrived in period 0, period 1 consists of both of their service times. \( C_1 \) is served until \( t = 8 \), and then \( C_2 \) is served until \( t = 11 \). During this period a single customer \( C_3 \) arrives, at \( t = 9 \). After \( C_2 \) completes service at \( t = 11 \), \( C_3 \) is served and period 2 begins. No customers arrive during this period, so the system becomes idle when \( C_3 \) leaves the system at \( t = 15 \); period 3 and all subsequent periods are empty.

We require two results related to the interval lengths \( S_j \).

**Lemma 3.3.1.** Let \( H_i(y) \) be the DF of the interval \( S_i \), with Laplace-Stieltjes transform \( \tilde{H}_i(s) \). Then for any \( i \in \mathbb{N} \),

\[
\tilde{H}_i(s) = \tilde{H}_{i-1}(\lambda - \lambda \tilde{B}(s)).
\]  

(3.37)

**Proof.** Let \( T_b \) be the duration of the entire busy period. Then

\[
T_b = \sum_{i=0}^{\infty} S_i.
\]  

(3.38)

Note that some \( S_m = 0 \), as eventually we will have an interval with no arrivals, and then for any \( n > m \), we have \( S_n = 0 \) also. We will show this in the next lemma; for now we are only interested in the relationship between consecutive intervals.

As before, we use conditioning to find the Laplace-Stieltjes transform of \( S_i \). We require the number of arrivals in the previous interval, which we denote by \( N_{i-1} \). Also, we denote the service times arrival \( j \) in period \( i - 1 \) by \( X_j \). Note that the \( X_j \) are IID, distributed as \( B \). Moreover, they are processed in interval \( i \), while we are conditioning on the length of interval \( i - 1 \), so they are independent of \( S_{i-1} \).
\[ E(e^{-sS_i}|S_{i-1} = y, N_{i-1} = n) = E(e^{\sum_{j=1}^{n}X_j}) = [\tilde{B}(s)]^n. \] (3.39)

It may seem unnecessary to condition on \( S_{i-1} = y \) above when the expression does not depend on \( y \). The reason is that the distribution of \( N_{i-1} \) is only tractable when conditioned upon \( S_{i-1} = y \). More specifically, the number of arrivals in a period of length \( y \) has a Poisson distribution with parameter \( \lambda y \).

\[ E(e^{-sS_i}|S_{i-1} = y) = \sum_{n=0}^{\infty} \frac{(\lambda y)^n}{n!} e^{-\lambda y}[\tilde{B}(s)]^n \] (3.41)

\[ = \exp(\lambda y(\tilde{B}(s) - 1)). \] (3.42)

Finally, we remove the conditioning on \( y \) to find a recursive expression for \( \tilde{H}_i(s) \).

\[ \tilde{H}_i(s) = E(e^{-sS_i}) \] (3.43)

\[ = \int_0^{\infty} \exp(\lambda y(\tilde{B}(s) - 1))dH_{i-1}(y) \] (3.44)

\[ = \tilde{H}_{i-1}(\lambda - \lambda \tilde{B}(s)). \] (3.45)

It can be helpful to view the busy period as a branching process. This is done by, amongst others, Wolff in [13], and Neuts [19], who bases his entire analysis of the M/G/1 on this insight.

**Lemma 3.3.2.** The number of arrivals to each interval of the busy period of an M/G/1 queue forms a branching process. The expected number of descendants of each interval is \( \rho \); thus in a stable queue the busy period duration is almost surely finite.

**Proof.** Let \( Z_n \) be the number of customers serviced in interval \( n \) of the busy period. The busy period starts with interval 0, when the initiating task is processed, so \( Z_0 = 1 \). Then each task \( i \) which is serviced in period \( n \) gives rise to some number of arrivals \( Y_{n,i} \), which will be serviced in period \( n + 1 \).

Because the arrivals form a Poisson process, the number of arrivals in an interval of length \( x \) has a Poisson distribution with parameter \( \lambda x \), so it must be a nonnegative integer. Moreover, the number of arrivals in disjoint intervals are independent. Now let \( X \) be the service time of arrival \( i \) to period \( n \).

\[ E(Y_{n,i}) = E(E(Y_{n,i}|X = x)) \] (3.46)

\[ = \lambda E(X) \] (3.47)

\[ = \rho. \] (3.48)
Thus the offspring distribution \( \{Y_{n,i} | n \in \mathbb{Z}_{\geq 0}, i \in \mathbb{N} \} \) is IID, and so \( \{Z_n | n \in \mathbb{Z}_{\geq 0} \} \) forms a branching process.

Moreover, by Theorem 2.3.1, if \( \rho < 1 \) the branching process dies out with probability 1. Hence, the busy period of a stable M/G/1 queue is finite with probability 1.

Now consider a customer who arrives during interval \( i \) of the busy period. Call their waiting time \( W \). As before, we denote the length of interval \( i \) by \( S_i \). We also define \( N \) as the number of arrivals in interval \( i \) prior to our customer, and \( Y \) as the residual lifetime of interval \( i \), or the amount of time from the arrival of our customer until the end of interval \( i \). Let the \( j^{th} \) customer to arrive in interval \( i \) before our customer have service time \( X_j \).

We also need to consider the possible states the queue could be in when our customer arrives. Let \( A_B \) be the event that our customer arrives to a busy system, and hence the event that our customer arrives to an idle system is \( A_B^C \). Also, let \( A_{B,i} \) be the event that our customer arrives to interval \( i \) of a busy period. Then \( \{A_{B,i}\}_{i=0}^\infty \) is a collection of disjoint sets, and

\[
\bigcup_{i=0}^{\infty} A_{B,i} = A_B. \tag{3.49}
\]

The waiting time of our customer is the sum of \( Y \), the remaining time in interval \( i \) (which is spent servicing customers who arrived in interval \( i - 1 \)), followed by part of interval \( i + 1 \), namely the service times of the \( N \) customers who arrived before our customer in interval \( i \).

We illustrate this in Figure 3.4. Consider the waiting time of customer \( C_3 \), who arrives during period 2 (the third period; recall that we commence counting from zero) of a busy period. Customers \( C_1, C_2, C_3 \) and \( C_4 \) arrive during this period, at times 12, 15, 17 and 19, respectively. As two customers arrived in interval 2 prior to \( C_3 \), \( N = 2 \). Note that we are not concerned with the customers who arrived in earlier intervals. Interval 2 ends at \( t = 20 \). Hence \( Y \), the residual lifetime of interval 2, is 3. At \( t = 20 \) interval 3 begins, and the customers who arrived during interval 2 are served. First, \( C_1 \) is served for 2 seconds, until \( t = 22 \), and then \( C_2 \) is served for 3 seconds, until \( t = 25 \). At this point \( C_3 \) can enter service. Thus, the entire waiting time for \( C_3 \) is 8; \( Y = 3 \), \( X_1 = 2 \), \( X_2 = 3 \). Interval 3 continues as \( C_3 \) and then \( C_4 \) are served, and there may be subsequent intervals if new customers arrive during interval 3, but none of these factors are relevant to the waiting time of \( C_3 \).

**Theorem 3.3.3.** The waiting time \( W \) for a customer in an M/G/1 FCFS queue has LST

\[
E(e^{-sw}) = \frac{s(1-\rho)}{s - \lambda + \lambda B(s)}. \tag{3.50}
\]
Proof.

\[ W = Y + \sum_{j=1}^{N} X_j \]  

\[ E(e^{-sW} | A_{B,i}, S_i = x, Y = y, N = n) = e^{-sy} + E(e^{-s\sum_{j=1}^{n} X_j}) \]

\[ = e^{-sy}[\tilde{B}(s)]^n. \]  

The \( N \) customers arrive in the portion of interval \( i \) prior to our customer, which has length \( x - y \). Thus \( N \) has a Poisson distribution with parameter \( \lambda(x - y) \) when conditioned on \( S_i = x, Y = y \).

\[ E(e^{-sW} | A_{B,i}, S_i = x, Y = y) = \sum_{n=0}^{\infty} P(N = n)E(e^{-sW} | A_{B,i}, S_i = x, Y = y, N = n) \]

\[ = e^{-sy} \sum_{n=0}^{\infty} \frac{(\lambda(x-y))^n}{n!} e^{-\lambda(x-y)[\tilde{B}(s)]^n} \]

Next we require the joint density of \( S_i \) and \( Y \). It is commonly stated (in particular, by Kleinrock in [6]) that this follows from renewal theory, and specifically from the limiting joint density of interval length \( A \) and lifetime \( Y \) which we gave in Equation (2.126). It is not entirely clear what the renewal process is, however; successive intervals of a busy period are not IID. It may be possible to argue that concatenating the \( j^{th} \) interval of each busy period together (assuming these intervals are non-empty) gives a renewal process, and that our conditioning gives an arrival to this system. We believe
that this theorem could be made both simpler and more rigorous by simply considering
Poisson arrivals to a given interval, however.

In any case, there is no dispute as to the actual density:

\[ f_{S_i,Y}(x,y) = \frac{f_{S_i}(x)y}{E(S_i)}, \quad 0 \leq y \leq x < \infty. \]  

(3.57)

Using the joint density, we can eliminate the conditioning on \( S_i \) and \( Y \).

\[ E(e^{-sW} | A_{B,i}) = \frac{1}{E(S_i)} \int_0^\infty \int_0^x \exp\{\lambda - s - \lambda \tilde{B}(s)\} y \exp\{\lambda \tilde{B}(s) - \lambda x\} dy \, dH_i(x) \]

\[ = \frac{1}{E(S_i)} \int_0^\infty \frac{e^{(\lambda \tilde{B}(s) - \lambda)x} - e^{-sx}}{s - \lambda + \lambda \tilde{B}(s)} dH_i(x) \]  

(3.58)

\[ = \tilde{H}_i(\lambda - \lambda \tilde{B}(s)) - \tilde{H}_i(s) \]  

(3.59)

\[ E(S_i)(s - \lambda + \lambda \tilde{B}(s)) \]  

(3.60)

However, from Lemma 3.3.1 we know that \( \tilde{H}_i(s) = \tilde{H}_{i-1}(\lambda - \lambda \tilde{B}(s)) \).

\[ E(e^{-sW} | A_{B,i}) = \frac{\tilde{H}_{i+1}(s) - \tilde{H}_i(s)}{E(S_i)(s - \lambda + \lambda \tilde{B}(s))}. \]  

(3.61)

Next, we need to know the probability that a customer who arrives during a busy
period arrives to the \( i \)th interval.

Poisson arrivals are independent of all queueing behaviour up prior to the arrival
time, and so one might intuitively expect that the probability of arriving to a given
state is equal to the proportion of time the system spends in that state. This is in fact
correct; it is known as the PASTA (Poisson arrivals see time averages) property, and
was first proven in general by Wolff in [20].

Recall that \( T_b \) is the length of the entire busy period.

\[ P(A_{B,i}|A_B) = \frac{E(S_i)}{E(T_b)}. \]  

(3.62)

Given this probability, we can find the conditional Laplace-Stieltjes transform of
waiting time for customers who arrive during a busy period.

\[ E(e^{-sW} | A_B) = \sum_{i=0}^{\infty} E(e^{-sW} | A_{B,i}) \frac{E(S_i)}{E(T_b)} \]  

(3.63)

\[ = \sum_{i=0}^{\infty} \frac{\tilde{H}_{i+1}(s) - \tilde{H}_i(s)}{E(T_b)(s - \lambda + \lambda \tilde{B}(s))}. \]  

(3.64)
By telescoping series and Lemma 3.3.2,
\[ E(e^{-sW}|A_B) = \frac{1 - \tilde{H}_0(s)}{E(T_b)(s - \lambda + \lambda \tilde{B}(s))}. \] (3.65)

We still need to factor in arrivals to an empty system, which clearly experience no waiting time.
\[ E(e^{-sW}) = (1 - \rho)E(e^{-sW}|A_B^c) + \rho E(e^{-sW}|A_B) \] (3.66)
\[ = 1 - \rho + \frac{\rho(1 - \tilde{H}_0(s))}{E(T_b)(s - \lambda + \lambda \tilde{B}(s))}. \] (3.67)

We can substitute the mean busy period length from Corollary (3.2.3) to determine the final expression for waiting time.
\[ E(e^{-sW}) = \frac{s(1 - \rho)}{s - \lambda + \lambda \tilde{B}(s)}. \] (3.68)

With some algebraic effort, we can derive the moments of waiting time from this expression. We will make considerable use of the expected waiting time in later sections, so we calculate it below.

**Theorem 3.3.4.** The waiting time \( W \) for a customer in an \( M/G/1 \) FCFS system has expectation
\[ E(W) = \frac{\lambda}{2\mu^{(2)}(1 - \rho)}. \] (3.69)

where \( 1/\mu^{(2)} \) is the second moment of the service time distribution \( B \).

**Proof.** As before, we differentiate the LST of a distribution function to find the mean.
\[ E(W) = -\lim_{s \to 0} \frac{\partial}{\partial s} E(e^{-sW}) \] (3.70)
\[ = -\lim_{s \to 0} \frac{(1 - \rho)(\lambda \tilde{B}(s) - \lambda s \tilde{B}'(s) - \lambda)}{(s - \lambda + \lambda \tilde{B}(s))^2}. \] (3.71)

This is an indeterminate form, so we use L'Hôpital's rule twice:
\[ E(W) = \lim_{s \to 0} \lambda (1 - \rho) \frac{s \tilde{B}''(s)}{2(s - \lambda + \lambda \tilde{B}(s))(\lambda \tilde{B}'(s) + 1)} \]  
(3.72)

\[ = \lim_{s \to 0} \lambda (1 - \rho) \frac{\tilde{B}''(s) + s \tilde{B}'''(s)}{2(1 + \lambda \tilde{B}'(s))^2}. \]  
(3.73)

Here we have eliminated the indeterminate forms, and the remaining Laplace-Stieltjes transforms of service time give moments of the service time distribution.

\[ E(W) = \frac{\lambda (1 - \rho)}{2 \mu^2 (1 - \lambda / \mu)^2} \]  
(3.74)

\[ = \frac{\lambda}{2 \mu^2 (1 - \rho)}. \]  
(3.75)

This is known as the Pollaczek-Khinchin formula for mean waiting time. It was originally derived by Pollaczek [21] using integral equations, and soon after by Khinchin [22] using probabilistic methods more familiar to modern queueing theory.

Observe that the mean waiting time depends only on the arrival rate and the first two moments of the service time. If we fix the mean service time and consider different service distributions, the mean waiting time increases with the variance of service time, and thus is minimised for deterministic service times.
Chapter 4

Waiting time distribution in an M/G/1 queue with absolute priorities

In this chapter, we modify the M/G/1 queue to allow for customer priorities. It is based on Conway, Maxwell and Miller’s analysis in [2].

As before, customers arrivals form a Poisson process, and the server processes a single customer at a time. If a customer arrives to a busy system, they are placed in a queue. However, we now have $P \in \mathbb{N}$ priority classes. Each customer has a priority $p \in \{1 \ldots P\}$; a lower value of $p$ indicates a higher priority. Whenever the server completes a job and multiple customers are queued, the highest priority customer will be chosen for service. Where multiple customers of the same priority class are queued, the first-come first-served discipline applies. The queue is non-preemptive; a high priority customer may not displace a low priority customer who has already commenced service.

We can consider arrivals of class $i$ customers to be generated by independent Poisson processes of rate $\lambda_i$. Equivalently, we can treat the system as having a single arrival process of rate $\lambda = \sum_{i=1}^{n} \lambda_i$, with an arriving customer having class $i$ with probability $\lambda_i/\lambda$.

The distribution of service time may vary by class; we denote the service time distribution function of class $p$ by $B_p$, with Laplace-Stieltjes transform $\tilde{B}_p(s)$ and mean $\mu_p^{-1}$. The Laplace-Stieltjes transform must exist in some neighbourhood of the origin. We define $\rho_p = \lambda_p/\mu_p$ to be the utilisation of customers of type $p$.

We are interested in the waiting time distribution for some customer class $k \in \{1 \ldots P\}$. The first observation we make is that, from the perspective of our class $k$ customer, all other classes can be reduced to two groups. Every customer of higher priority than $k$ will overtake a class $k$ customer whenever they are queued together, and every customer with lower priority than $k$ will be overtaken by a class $k$ customer. In fact, the only time that a lower priority customer will affect a class $k$ customer at all is if the class $k$ customer arrives when the lower priority customer is already in service.

Thus, we assign all high priority customers (those with priorities in $1 \ldots k - 1$) to
class $H$, and low priority customers (priorities in $k+1 \ldots P$) to class $L$. These customers arrive according to independent Poisson processes of rates $\lambda_H$ and $\lambda_L$ respectively, where

$$\lambda_H = \sum_{p=1}^{k-1} \lambda_p, \quad (4.1)$$

$$\lambda_L = \sum_{p=k+1}^{P} \lambda_p. \quad (4.2)$$

The service time of a class $H$ or $L$ customer is a mixture of the service times of their underlying priority classes. If we denote the service time of some high priority customer by $X_H$, and the event that this customer has priority $p \in \{1 \ldots k-1\}$ by $A_p$, we find that

$$B_H(x) = P(X_H \leq x) \quad (4.3)$$

$$= \sum_{p=1}^{k-1} P(X_H \leq x | A_p) P(A_p) \quad (4.4)$$

$$= \sum_{p=1}^{k-1} \lambda_p \lambda_H B_p(x). \quad (4.5)$$

The same reasoning shows that

$$B_H(x) = \sum_{p=k+1}^{P} \frac{\lambda_p}{\lambda_L} B_p(x). \quad (4.6)$$

By the linearity of expectation (or integration), the same mixing procedure applies to the Laplace-Stieltjes transforms of the service times.

$$\tilde{B}_H(z) = \sum_{p=1}^{k-1} \frac{\lambda_p}{\lambda_H} \tilde{B}_p(x), \quad (4.7)$$

$$\tilde{B}_L(z) = \sum_{p=k+1}^{P} \frac{\lambda_p}{\lambda_L} \tilde{B}_p(x). \quad (4.8)$$

Note that this analysis allows for $k = 1$ and $k = P$ — the highest and lowest priorities, respectively. In these cases either class $H$ or class $L$ will not exist; they will have zero arrival rate and undefined service distribution.
4.1 Delay cycles

In this section we consider particular patterns of customers, known as delay cycles. We determine the Laplace transform of the distribution of a delay cycle, and the waiting time for a customer arriving while the system is in a delay cycle. We use these distributions to determine the waiting time distribution for an M/G/1 queue with a first-come first-served discipline, and with absolute priorities. The analysis is based on [2] and [6].

Returning to our classification of customers into types $k$, $H$, and $L$, consider a situation where a class $k$ customer $C_1$ is in service and another class $k$ customer $C_2$ is queued. In this case, any class $H$ customer who arrives during $C_1$’s service will be served before $C_2$. Furthermore, additional class $H$ customers may arrive while the first class $H$ customer is in service, more may arrive while those are in service, and so on. This situation is identical to the busy periods considered in the previous chapter, but with arrival rates and service times restricted only to high priority customers; observe that lower priority arrivals will not be served until the system is free of class $H$ customers. Thus, by Theorem 3.2.2 we have a functional equation for $G_{H^+}$, the Laplace-Stieltjes transform of the length of a busy period of class $H$ customers.

$$\tilde{G}_{H^+}(s) = \tilde{B}_H(s + \lambda_H - \lambda_H \tilde{G}_{H^+}(s)).$$  (4.9)

We have appropriated some notation from computer science in this chapter; $H^+$ indicates that the server will process at least one $H$ job, and $H$ jobs may be repeated an arbitrary number of times before some other job can be served. We believe that this notation clarifies matters relative to our source material, where, for example, $\gamma_a$ is the distribution of service time for a class $a$ job and $\eta_a$ is a busy period of class $a$, but a cycle of one class $k$ job followed by arbitrarily many class $a$ blocking tasks is $\gamma_{ka}$. [2, pp. 160-161] In general, it can be rather difficult to keep track of whether we are considering a single task or a busy period of such tasks.

If we want to know the delay from when $C_1$ enters service until $C_2$ enters service in the example above, we need a little more than just a busy period of class $H$ customers. Firstly, we must consider the service time of $C_1$. Secondly, the number of high priority busy periods between $C_1$ and $C_2$ is uncertain. There may be no class $H$ arrivals (and note that there can be none queued when $C_1$ enters service, or they would be served first). Alternatively, there may be any number of class $H$ arrivals while $C_1$ is served, each of which will give rise to its own busy period. We describe the list of jobs as $kH^*$; the asterisk indicates that there may be any number of class $H$ jobs, including zero.

It is worth noting that these task orders have been chosen so that they will not be interrupted. In particular, it does not make sense to consider a busy period of class $k$ or $L$ jobs, because any class $H$ arrival would interrupt the sequence.

If we carefully choose our task orders, we can construct more complicated sequences of jobs, which we refer to as delay cycles. For example, we can consider an interval in which the system repeatedly processes class $H$ and $k$ jobs. Assume that the first customer to be served, $C_1$, is of class $k$. While $C_1$ is in service some class $H$ jobs
may arrive; if so there will be a busy period of class $H$ jobs before the second class $k$ customer, $C_2$, may be served. The same applies while $C_2$ is in service, and so on, until no class $H$ or $k$ jobs are in service.

In general, a delay cycle is made up of an initial delay, followed by the processing of zero or more blocking tasks. The cycle ends when the initial delay has finished, all blocking tasks are finished and there are no further blocking tasks in the system. We denote the duration of a delay cycle by $T_{IB}$, continuing our notation from the previous section. It is worth noting here that Conway, Maxwell and Miller’s definitions of delay cycles and blocking tasks are a little unclear; they claim that a delay cycle ends with an idle system [2, p. 151], but later contradict themselves by allowing delay cycles to end when only low-priority jobs are queued [2, p. 161]. They are also a little unclear on exactly what can constitute the jobs within a delay cycle. We believe that our definitions of delay cycles and blocking tasks clarify their intentions.

We show a sample delay cycle in Figure 4.1. The cycle is of type $kH^*$ — the initial delay is a class $k$ customer, and subsequent blocking tasks are type $H$ customers. Firstly, a class $k$ customer enters service at time zero. While this customer is in service, high priority customers $C_1$ and $C_2$ arrive at $t = 3$ and $t = 5$, as well as low priority customer $C_3$ at $t = 6$. Our cycle does not include low priority customers, so $C_3$ will not be processed until after our cycle ends. After our initial delay is over, $C_1$ is served, with no customers arriving during its service. Then $C_2$ is served from $t = 10$ to $t = 12$, with high priority customer $C_4$ arriving at $t = 11$. Thus $C_4$ is served from $t = 12$ to $t = 14$, with class $k$ customer $C_5$ arriving at $t = 13$. At $t = 14$, there are no remaining high priority customers, so the delay cycle ends. Note that $C_3$ and $C_5$ are still queued, so the system will not become idle; instead another class $kH^*$ cycle will begin.

In this example, blocking tasks are single type $H$ customers, but in general a blocking task consists of one or more customers who are processed sequentially. The blocking task must represent a single interval of time - the server may not pause the processing of
one blocking task in order to serve some other customer. Moreover, blocking tasks must be processed on a first-come, first-served basis. This does not preclude the application of delay cycles to priority queues, as we shall see.

The duration of a blocking task has distribution $G_B$ and Laplace-Stieltjes transform $\tilde{G}_B(z)$. Because blocking tasks are initiated by some subset of customers, who arrive as a Poisson process, the arrival of blocking tasks themselves form a Poisson process of rate $\lambda_B$. Moreover, because of the memoryless arrivals and independent service times, the durations of multiple blocking tasks are IID.

An initial delay must satisfy the same conditions as a blocking task, but within a given delay cycle the specific combination of jobs which make up an initial delay can differ from the blocking tasks, as in our previous example. Its duration is given by $T_I$.

### 4.1.1 Distribution of delay cycle length

As in chapter 3, we observe that the length of an entire delay cycle is independent of the queueing discipline used to prioritise blocking tasks within the cycle [6]. Therefore, we can use an alternative discipline if it makes the system easier to analyse. For this section, we will assume that blocking tasks are processed in a last-in-first-out manner. In other words, when the system finishes processing a blocking task, it will select the most recently queued blocking task to service next.

An example $kH^*$ cycle is shown in Figure 4.2. The cycle begins when a class $k$ customer commences service at $t = 0$. This initial customer remains in service until $t = 7$, with high priority customers $C_1$ and $C_2$ arriving at $t = 3$ and $t = 5$, and low priority customer $C_3$ arriving at $t = 6$. After the initial customer, $C_2$ is served from $t = 7$ to $t = 9$ — recall that we are assuming a LCFS discipline here — with no arrivals in this interval. Then $C_1$ is served from $t = 9$ to $t = 12$, with a high priority customer...
$C_4$ arriving at $t = 11$. Then $C_4$ is served from $t = 12$ to $t = 14$, with a class $k$ customer arriving at $t = 13$. At $t = 14$, there are no further class $H$ customers in the system, so the delay cycle ends; the overall length is thus 14 seconds.

We now consider a more general result.

**Theorem 4.1.1.** The length of a generalised delay cycle has LST

$$\tilde{G}_{IB^*}(s) = \tilde{G}_I(s + \lambda_B - \lambda_B\tilde{G}_{B^+}(s))$$ \hspace{1cm} (4.10)

where $G_I$ is the DF of the initial delay, $\lambda_B$ is the arrival rate of blocking tasks, and $G_{B^+}$ is the DF of a busy period of blocking tasks, which satisfies

$$\tilde{G}_{B^+}(s) = \tilde{G}_B(s + \lambda_B - \lambda_B\tilde{G}_{B^+}(s)).$$ \hspace{1cm} (4.11)

**Proof.** Let the number of blocking tasks which arrive during the initial delay be $N$, and the initial delay be $T_I$, with distribution $G_I$. Each blocking task generates its own busy period, as additional blocking tasks can arrive while the first in service. We denote the duration of the busy period of the $i^{th}$ blocking task by $T_i$. Because each busy period consists of a distinct set of customers and are started by the same type of task, their durations are IID. We denote their common distribution by $G_{B^+}$.

The duration of the delay cycle is exactly that of the initial delay, plus the processing time of each busy period initiated by a blocking task arriving during the initial delay. Thus

$$E(e^{-sT_{IB^*}}|N = n, T_I = t_0, X_i = x_i) = \exp(-s(t_0 + \sum_{i=1}^{n} x_i))$$ \hspace{1cm} (4.12)

$$E(e^{-sT_{IB^*}}|N = n, T_I = t_0) = e^{-st_0}E\left(\prod_{i=1}^{n} e^{-sx_i}\right)$$ \hspace{1cm} (4.13)

$$= e^{-st_0}\tilde{G}_{B^+}(s)^n.$$ \hspace{1cm} (4.14)

The arrival of blocking tasks forms a Poisson process with rate $\lambda_B$, allowing us to remove the conditioning on $N$.

$$P(N = n|T_I = t_0) = \frac{e^{\lambda_Bt_0}((\lambda_Bt_0)^n}{n!}$$ \hspace{1cm} (4.15)

$$E(e^{-sT_{IB^*}}|T_I = t_0) = \sum_{n=0}^{\infty} \frac{e^{-\lambda_Bt_0}((\lambda_Bt_0)^n}{n!}e^{-st_0}\tilde{G}_{B^+}(s)^n$$ \hspace{1cm} (4.16)

$$= \exp(-t_0(s + \lambda_B - \lambda_B\tilde{G}_{B^+}(s))).$$ \hspace{1cm} (4.17)
Finally, recall that the initial delay $T_I$ has distribution function $G_I$, with Laplace transform $\tilde{G}_I(s)$, giving us the unconditional Laplace transform

$$
\tilde{G}_{IB^+}(s) = E(e^{-sT_{IB^+}}) = \int_0^\infty \exp(-t(s + \lambda_B - \lambda_B\tilde{G}_{B^+}(s)))dG_I(t) = \tilde{G}_I(s + \lambda_B - \lambda_B\tilde{G}_{B^+}(s)).
$$

From here we require the LST $\tilde{G}_{B^+}(s)$. It can be found either by using Equation (4.20) above with $I = B$, or Theorem 3.2.2 giving the busy period duration in a FCFS queue. Recall that nothing will interrupt the blocking tasks until the system is free of them, and so we can ignore other jobs and treat the priority system as FCFS. Both result in the same functional equation,

$$
\tilde{G}_{B^+}(s) = \tilde{G}_B(s + \lambda_B - \lambda_B\tilde{G}_{B^+}(s)).
$$

As before, for particularly simple blocking task distributions this may be solved analytically, but in general we must solve it numerically. We can, however, analytically determine the moments of the busy period and delay cycle durations from Theorem 4.1.1.

**Theorem 4.1.2.** Consider a general busy period, in which blocking tasks have utilisation $\rho_B$ and mean $\mu_B^{-1}$. The expected length of such a busy period is

$$
\frac{1}{\mu_{B^+}} = \frac{1}{\mu_B(1 - \rho_B)}.
$$

If we have a general delay cycle whose initial delay has mean $\mu_I^{-1}$, the cycle has mean length

$$
\frac{1}{\mu_{IB^+}} = \frac{1}{\mu_I(1 - \rho_B)}.
$$

**Proof.** We simply differentiate the Laplace-Stieltjes transforms from Theorem 4.1.1.
\[ \frac{1}{\mu_{B^+}} = -\tilde{G}_{B^+}'(0) \]  
\[ = -(1 - \lambda_B \tilde{G}_{B^+}'(0))(\frac{\mu_B}{1 + \lambda_B \mu_{B^+}^{-1}}) \tilde{G}'(0) \]  
\[ = \frac{1}{\mu_B(1 - \rho_B)}. \]  
(4.24)  
(4.25)  
(4.26)  
(4.27)  
(4.28)

Recall from Section 2.1 that any LST \( \tilde{F} \) satisfies \( \tilde{F}(0) = 1 \). The utilisation \( \rho_B \) is simply the product of the arrival rate and the expected duration of blocking tasks.

The expected delay cycle length is found in a similar manner.

\[ \frac{1}{\mu_{IB^*}} = -\tilde{G}_{IB^*}'(0) \]  
\[ = -(1 - \lambda_B \tilde{G}_{IB^*}'(0))(\frac{\mu_I}{1 + \frac{\rho_B}{1 - \rho_B}}) \tilde{I}'(0) \]  
\[ = \frac{1}{\mu_I(1 - \rho_B)}. \]  
(4.29)  
(4.30)  
(4.31)  
(4.32)  
(4.33)

\[ \Box \]

### 4.1.2 Distribution of waiting time for arrivals in a delay cycle

In this section we consider the waiting time for a customer who arrives during a delay cycle. The customer must be the first in a blocking task. When we apply these results in later sections, we will construct appropriate delay cycles and blocking tasks in order to determine the waiting time distribution for any customer.

We now divide the delay cycle into intervals a recursive fashion. The first interval is just the initial delay. The next interval consists of the service of each blocking task to arrive during the initial delay, and so on. Thus, every task to arrive during \( n^{th} \) interval is served during the \( n + 1^{th} \) interval. We denote the duration of the \( n^{th} \) interval by \( S_n \). Note that if no blocking tasks arrive to the \( n^{th} \) interval, then the \( n + 1^{th} \) interval will be empty, and thus every successive interval will also be empty.

This is illustrated in Figure 4.3, which shows the same sequence of arrivals as the Figure 4.2; we are again considering a \( kH^* \) cycle. The initial delay is the service
duration of the initial class $k$ customer, which has length $T_I$. Two type $H$ customers $C_1$ and $C_2$ arrive during the initial delay; they are served during the next interval, of length $T_1$. During this interval, one more type $H$ customer $C_4$ arrives; this customer is served in the following interval, of length $T_2$. No type $H$ customers arrive during this interval, so the delay cycle ends after this. Note that one class $k$ and one type $L$ customer arrive during this cycle, but will not be served until after it concludes, as we only consider high priority customers as blocking tasks. (Note that the figure also shows the waiting time of $C_2$; we will consider this later.)

Using this partition, we can consider the length of the entire delay cycle:

\[ T_{IB^*} = \sum_{j=0}^{\infty} S_j. \] (4.34)

**Theorem 4.1.3.** Let $W$ be the waiting time of a typical customer who arrives during a delay cycle. Then the LST of $W$ satisfies

\[ E(e^{-sW}) = \frac{\mu_B(1 - \rho_B)(\bar{G}_I(s) - 1)}{\lambda_B - \lambda_B\bar{G}_B(s) - s}. \] (4.35)

**Proof.** Let $\sigma_j$ be the event that the customer arrived to the $j^{th}$ interval. Because we are only considering arrivals to this delay cycle, $\{\sigma_j : j \in \mathbb{N}\}$ forms a partition of our
sample space. Let the number of blocking tasks to arrive during the $j^{th}$ interval prior to our customer be $N$, and the remaining time in this interval be $Y$.

Recall that each task from the $j^{th}$ interval is processed in the $(j+1)^{th}$ interval. Moreover, blocking tasks are processed in order of their arrival. Thus, after our customer arrives they must wait until the $j^{th}$ interval concludes, and then for each of the $N$ blocking tasks to be processed in the $(j+1)^{th}$ interval.

\[ E(e^{-sW|\sigma_j, S_j = t, Y = y, N = n, X_i = x_i}) = e^{-s(y + \sum_{i=1}^{n} x_i)}. \quad (4.36) \]

We show an example of this in Figure 4.3. Consider the waiting time of customer $C_2$, who arrives at $t = 5$ during the initial delay (interval 0). The initial delay lasts until $t = 7$, so in this case $Y = 2$. A single type $H$ customer, $C_1$, arrives during the initial delay and before $C_2$. Thus, $N = 1$. This customer $C_1$ is in service from $t = 7$ to $t = 10$; thus $X_1 = 3$. After this, $C_2$ commences service. Thus, in this case the total delay is $Y + X_1 = 5$.

Because the $N$ blocking tasks which arrive before our customer in the $j^{th}$ interval are processed during the $(j+1)^{th}$ interval, their processing times $\{X_i\}_{i=1}^{N}$ are independent of the other variables on which we are conditioning. Moreover, they are independent of each other, with distribution function $G_B$.

\[ E(e^{-sW|\sigma_j, S_j = t, Y = y, N = n}) = e^{-sy}E(e^{-s\sum_{i=1}^{n} X_i}) = e^{-sy}\tilde{G}_B(s)^n. \quad (4.38) \]

The arrival of blocking tasks forms a Poisson process with parameter $\lambda_B$. There is a period of length $t - y$ in $I_j$ before our customer arrives. Hence we can determine the distribution of $N$ and remove the conditioning as follows:

\[ P(N = n|S_j = t, Y = t) = e^{-\lambda_B(t-y)}\frac{(\lambda_B(t-y))^n}{n!} \quad (4.39) \]

\[ E(e^{-sW|\sigma_j, S_j = t, Y = y}) = \sum_{n=0}^{\infty} e^{-\lambda_B(t-y)}\frac{(\lambda_B(t-y))^n}{n!}e^{-sy}\tilde{G}_B(s)^n \]
\[ = e^{-\lambda_B(t-y)-sy}e^{\lambda_B\tilde{G}_B(s)(t-y)}. \quad (4.40) \]

As in the previous chapter, we require the joint distribution of $S_j$ and $Y$, conditional on arrival during a delay cycle in state $j$. The density is identical:

\[ f_{Y,S_j}(y,t) = \frac{1}{E(S_j)} \frac{dH_j(t)}{dt}, t > 0, 0 < y < t. \quad (4.42) \]

Using the joint distribution of $Y$ and $S_j$ we can remove the conditioning on those variables in Equation (4.41):
\[ E(e^{-sW|\sigma_j}) = \int_0^\infty \int_0^t \exp(\lambda_B y - \lambda_B t - sy + \lambda_B \tilde{G}_B(s)(t - y)) \frac{1}{E(S_j)} dy dH_j(t) \]  
(4.43)

\[ = \frac{1}{E(S_j)} \int_0^\infty \exp(-t(\lambda_B - \lambda_B \tilde{G}_B(s)))dH_j(t) \int_0^t \exp(y(\lambda_B - \lambda_B \tilde{G}_B(s) - s))dy \]  
(4.44)

\[ = \frac{1}{E(S_j) \lambda_B - \lambda_B \tilde{G}_B(s) - s} \int_0^\infty \exp(-ts) - e^{-\lambda_B - \lambda_B \tilde{G}_B(s)t} dH_j(t). \]  
(4.45)

\[ = \frac{1}{E(S_j)(\lambda_B - \lambda_B \tilde{G}_B(s) - s)} \int_0^\infty e^{-ts} - e^{-\lambda_B - \lambda_B \tilde{G}_B(s)t} dH_j(t). \]  
(4.46)

\[ = \frac{1}{E(S_j)(\lambda_B - \lambda_B \tilde{G}_B(s) - s)} \int_0^\infty e^{-ts} - e^{-\lambda_B - \lambda_B \tilde{G}_B(s)t} dH_j(t). \]  
(4.47)

To progress from here we need \( H_j(t) \) — the distribution of \( S_j \), the length of the \( j \)th interval. Recall that for positive \( j \), \( S_j \) is the time to service each blocking task to arrive during the \( j - 1 \)th interval. Let \( N_j \) be the number of such arrivals, and \( Z_1, \ldots, Z_n \) be their durations. As always, the blocking task durations are IID, with common distribution function \( G_B \). Then

\[ E(e^{-sS}|S_{j-1} = t, N_{j-1} = n) = E(e^{s \sum_{i=1}^n X_i}) \]  
(4.48)

\[ = \prod_{i=1}^n E(e^{sX_i}) \]  
(4.49)

\[ = \tilde{G}_B(s)^n. \]  
(4.50)

Because blocking tasks arrive according to a Poisson process with rate \( \lambda_B \), we know the distribution of \( N_{j-1} \) given \( S_{j-1} \), and can remove the conditioning on \( N_{j-1} \) in Equation (4.48).

\[ P(N_{j-1} = n|S_{j-1} = t) = e^{-\lambda_B t} \frac{(\lambda_B t)^n}{n!}. \]  
(4.51)

\[ \therefore E(e^{-sS}|S_{j-1} = t) = \sum_{n=0}^\infty e^{-\lambda_B t} \frac{(\lambda_B t)^n}{n!} \tilde{G}_B(s)^n \]  
(4.52)

\[ = e^{-\lambda_B t + \tilde{G}_B(s) \lambda_B t}. \]  
(4.53)

Recall that \( H_j \) is the CDF of \( S_j \), with Laplace-Stieltjes transform \( \tilde{H}_j \). We can thus remove the conditioning on \( S_{j-1} \) to get a recurrence relation for \( \tilde{H}_j(s) \):
\[
\tilde{H}_j(s) = E(e^{-sS_j}) = \int_0^\infty e^{-\lambda_B t + \tilde{G}_B(s)\lambda_B t} dH_{j-1}(t) = \tilde{H}_{j-1}(\lambda_B - \lambda_B \tilde{G}_B(s)).
\]

Substituting our recurrence relation for \(\tilde{H}_j\) into Equation (4.47) we find
\[
E(e^{-sW} | \sigma_j) = \frac{\tilde{H}_j(s) - \tilde{H}_{j+1}(s)}{E(S_j)(\lambda_B - \lambda_B \tilde{G}_B(s) - s)}.
\]

We must now consider the probability that the customer arrives during a given interval \(I_j\), given that the customer arrives during a delay cycle. As in the previous chapter, this probability is equal to the proportion of time that a busy system spends in interval \(j\), by the PASTA property [20]. (We are only considering waiting time for customers who arrive during a delay cycle here. Clearly, if the customer arrives to an empty system, the waiting time will be zero.) Hence
\[
P(\sigma_j) = E(S_j)\mu_{IB^*}.
\]

Now we can remove the conditioning on \(j\) in Equation (4.57).
\[
E(e^{-sW}) = \sum_{j=0}^\infty P(\sigma_j) \frac{\tilde{H}_j(s) - \tilde{H}_{j+1}(s)}{E(S_j)(\lambda_B - \lambda_B \tilde{G}_B(s) - s)}.
\]
\[
= \frac{\mu_{IB^*}}{\lambda_B - \lambda_B \tilde{G}_B(s) - s} \sum_{j=0}^\infty \tilde{H}_j(s) - \tilde{H}_{j+1}(s)
\]
\[
= \frac{\mu_{IB^*}(\tilde{H}_0(s) - \tilde{H}_{\infty}(s))}{\lambda_B - \lambda_B \tilde{G}_B(s) - s}.
\]

Using a similar branching processes argument to the previous chapter, we can show that the \(S_i\) are eventually zero with probability 1, as long as \(\rho_B < 1\). Moreover, the first interval \(S_0\) is simply the initial delay; hence \(\tilde{H}_0(s) = \tilde{G}_I(s)\), and we can simplify the last expression to
\[
E(e^{-sW}) = \frac{\mu_{IB^*}(\tilde{G}_I(s) - 1)}{\lambda_B - \lambda_B \tilde{G}_B(s) - s}.
\]

Next, we substitute in \(\mu_{IB^*}\) from Theorem 4.1.2:
\[
E(e^{-sW}) = \frac{\mu_I(1 - \rho_B)(\tilde{G}_I(s) - 1)}{\lambda_B - \lambda_B \tilde{G}_B(s) - s}.
\]
We can also expand $\rho_B$ and $E(T_I)$ in terms of the other parameters. Thus, we only need to know $\lambda_B, \tilde{G}_I(s)$ and $\tilde{G}_B(s)$ — the arrival rate of blocking tasks, the initial delay distribution and the blocking time distribution, respectively — in order to determine the waiting time distribution.

$$E(e^{-sW}) = \frac{(1 + \lambda_B\tilde{G}_B'(0))(1 - \tilde{B}_0(s))}{\tilde{B}_0'(0)(\lambda_B - \lambda_B\tilde{G}_B(s) - s)}.$$  \hspace{1cm} (4.64)

### 4.2 Classifying queue states

The previous section gives us the waiting time distribution for an arrival to a specified delay cycle. In order to determine an overall waiting time, we need to categorise the operation of the $M/G/1$ absolute priority queue into different delay cycles. Following Conway, Maxwell and Miller [2], we give three such delay cycles, which we call type $H$, $k$ and $L$, after the first customer in each. The queue is always either idle, or in one of these delay cycles.

Recall that a delay cycle is specified by an initial delay and then a busy period of blocking tasks. Each cycle has identical blocking tasks — they commence when a class $k$ job enters service, and continue until the initial class $k$ job has completed and there are no more class $H$ jobs in the system. Following our convention, we denote the duration of one of these blocking tasks by $X_{kH^*}$.

Each class $k$ job arriving to a busy system initiates one of these blocking tasks; hence the rate at which blocking tasks arrive is $\lambda_k$. Note that there no class $H$ job can be in the system at the beginning of the blocking task, or it would be processed in preference to the class $k$ job. The class $H$ jobs involved in this blocking task are only those which arrive after the class $k$ customer enters service.

A type $H$ cycle starts when a class $H$ job arrives to an empty system. The initial delay is the time for the system to clear all class $H$ jobs, and subsequent blocking tasks are as described above.

A type $k$ cycle starts when a class $k$ job arrives to an empty system. The initial task is the time for the initial job to finish, and the system to finish any queued class $H$ jobs — in this case, the initial delay is of the same form as the blocking task, so it could also be considered as a busy period of class $kH^*$ blocking tasks. Type $H$ and $k$ cycles are shown in Figure 4.4 - observe that the only difference is the initial job in the cycle.

A type $L$ cycle starts whenever a class $L$ job enters service, not necessarily by arriving to an empty system — the system might instead have been processing class $H$ or $k$ jobs previously. The initial delay is again the time required to process the class $L$ arrival, and clear the system of class $H$ jobs. Type $L$ cycles are shown separately in Figure 4.4.

Observe that the above definitions ensure that when the system is busy, it is always in exactly one of the above delay cycle types. To see this, assume that the system is
System free of type H jobs

Initial delay Single blocking task

Type H or k job
arrives to idle system

Type H jobs
until none left

Type k job
(if any queued)

Type H jobs
until none left

$kH^*$ cycles
until none left

System free of type H and k jobs

Overall type H or k delay cycle

Type L job
begins service

System free of type H jobs

System free of type H jobs

System free of type H and k jobs

Overall type L delay cycle

Figure 4.4: Type H, k and L cycles
busy. If it commenced serving a class \( L \) customer more recently than it was last idle, it is in a type \( L \) cycle. Otherwise, the last customer to arrive to the system while it was idle was either of class \( H \) or \( k \), and the system is in either a type \( H \) or \( k \) cycle, respectively.

### 4.2.1 Blocking times and initial delay durations

Having divided the operation of our queue into delay cycles, we now apply our general delay cycle results to specific cycles of interest to us. Note that the overall type \( H \), \( k \) and \( L \) cycles are composed of smaller cycles, so we begin by considering the length of each component of our larger cycles.

The blocking time \( X_{kH^*} \) consists of the service of a single class \( k \) customer as the initial delay, and subsequent blocking times are busy periods consisting only of type \( H \) customers. Type \( H \) customers arrive at rate \( \lambda_H = \sum_{i=1}^{k-1} \lambda_i \).

We denote the Laplace-Stieltjes transform of \( X_{kH^*} \) by \( \tilde{B}_{kH^*}(s) \); by Theorem 4.1.1 it satisfies the functional equation

\[
\tilde{G}_{kH^*}(s) = \tilde{B}_k(s + \lambda_H - \lambda_H \tilde{G}_{H^+}(s)). \tag{4.65}
\]

Recall from Equation (4.9) that \( \tilde{G}_{H^+}(s) \) is the LST of a busy period of type \( H \) customers, satisfying

\[
\tilde{G}_{H^+}(s) = \tilde{B}_H(s + \lambda_H - \lambda_H \tilde{G}_{H^+}(s)). \tag{4.66}
\]

The initial delay \( \tilde{B}_H(s) \) above is the duration of a single class \( H \) service, which we considered earlier in Equation (4.7).

We can also use Theorem 4.1.2 to determine the expected blocking time, noting again that the initial delay is the service time for a single class \( k \) customer, and the blocking tasks are single type \( H \) customers. The utilisation of blocking tasks follows.

\[
\frac{1}{\mu_{kH^*}} = \frac{1}{\mu_k(1 - \rho_H^*).} \tag{4.67}
\]

\[
\rho_{kH^*} = \frac{\lambda_k}{\mu_{kH^*}} = \frac{\rho_k}{1 - \rho_H^*}. \tag{4.68}
\]

Note that \( \rho_H \) is the utilisation of all type \( H \) customers, or the proportion of the time the system spends servicing type \( H \) customers. The latter definition implies that it satisfies

\[
\rho_H = \sum_{i=1}^{k-1} \rho_i. \tag{4.70}
\]
The initial delays in type \( H \) and \( k \) cycles are familiar. For type \( H \) cycles, the initial delay is a busy period of type \( H \) customers, with distribution \( \tilde{G}_{H^+}(s) \) as used above. The initial delay in a type \( k \) cycle is another blocking time \( X_{kH^+} \), with distribution \( \tilde{G}_{kH^+}(s) \).

For type \( L \) cycles, the initial delay is a \( LH^* \) cycle; the initial task is a type \( L \) customer, while blocking times are busy periods of type \( H \) customers. As with \( kH^* \) cycles, blocking tasks have duration \( \tilde{B}_{H^+}(s) \) and arrival rate \( \lambda_H \). By Theorem 4.1.1 the overall duration has distribution

\[
\tilde{G}_{LH^*}(s) = \tilde{B}_L(s + \lambda_H - \lambda_H \tilde{G}_{H^+}(s)).
\]  

(4.71)

Theorem 4.1.2 gives us the mean length of each initial delay.

\[
\frac{1}{\mu_{H^+}} = \frac{1}{\mu_H(1 - \rho_H)};
\]  

(4.72)

\[
\frac{1}{\mu_{kH^*}} = \frac{1}{\mu_k(1 - \rho_H)};
\]  

(4.73)

\[
\frac{1}{\mu_{LH^*}} = \frac{1}{\mu_L(1 - \rho_H)}.
\]  

(4.74)

### 4.2.2 Delay cycle properties

We can now determine the waiting time distribution for a customer, given the type of delay cycle the system is processing on their arrival. We denote the Laplace-Stieltjes transforms of waiting time for customers arriving during \( H \), \( k \) and \( L \) type cycles as \( \tilde{W}_{HC}(s) \), \( \tilde{W}_{kC}(s) \) and \( \tilde{W}_{LC}(s) \), respectively.

This requires several parameters. Firstly, blocking tasks arrive to each delay cycle at rate \( \lambda_k \) and have length distribution \( \tilde{G}_{kH^*}(s) \). The initial delays were determined previously to be \( \tilde{G}_{H^+}(s) \), \( \tilde{G}_{kH^+}(s) \) and \( \tilde{G}_{LH^+}(s) \), for type \( H \), \( k \) and \( L \) cycles, respectively. Moreover, their expectations are \( 1/(\mu_H(1 - \rho_H)) \), \( 1/(\mu_k(1 - \rho_H)) \) and \( 1/(\mu_L(1 - \rho_H)) \). The utilisation of blocking tasks is given in Equation (4.69) as \( \rho_k/(1 - \rho_H) \).

Substituting these parameters into Equation (4.63) shows that the LST of waiting time for a customer arriving during a specified delay cycle is given by

\[
\tilde{W}_{HC}(s) = \frac{\mu_H(1 - \rho_H - \rho_k)(\tilde{G}_{H^+}(s) - 1)}{\lambda - \lambda \tilde{G}_{kH^+}(s) - s},
\]  

(4.75)

\[
\tilde{W}_{kC}(s) = \frac{\mu_k(1 - \rho_H - \rho_k)(\tilde{G}_{kH^+}(s) - 1)}{\lambda - \lambda \tilde{G}_{kH^+}(s) - s},
\]  

(4.76)

\[
\tilde{W}_{LC}(s) = \frac{\mu_L(1 - \rho_H - \rho_k)(\tilde{G}_{LH^+}(s) - 1)}{\lambda - \lambda \tilde{G}_{kH^+}(s) - s}.
\]  

(4.77)
The expected length of a delay cycle is given by Theorem 4.1.2; we need the expected length of the initial delays from the previous section and the utilisation of blocking tasks from Equation (4.69).

\[
\frac{1}{\mu_{HC}} = \frac{1}{\mu_H(1 - \rho_Hk^*)}, \quad (4.78)
\]

\[
\frac{1}{\mu_{kC}} = \frac{1}{\mu_k(1 - \rho_H - \rho_k)}, \quad (4.79)
\]

\[
\frac{1}{\mu_{LC}} = \frac{1}{\mu_L(1 - \rho_H - \rho_k)}. \quad (4.80)
\]

\[
\frac{1}{\mu_{HC}} = \frac{1}{\mu_H(1 - \rho_H - \rho_k^*)}, \quad (4.79)
\]

\[
\frac{1}{\mu_{kC}} = \frac{1}{\mu_k(1 - \rho_H - \rho_k)}, \quad (4.80)
\]

\[
\frac{1}{\mu_{LC}} = \frac{1}{\mu_L(1 - \rho_H - \rho_k)}. \quad (4.81)
\]

\[
4.2.3 \text{ Distribution of queue states}
\]

We have determined the distribution of waiting time for an arrival to each delay cycle. Moreover, the waiting time for an arrival to an empty system is zero, meaning it has Laplace-Stieltjes transform identically equal to one. Hence if \(\pi_H, \pi_k, \pi_L, \pi_0\) are the proportion of time that the system spends in \(H, k, L\) cycles and idle periods, the overall LST of waiting time is given by

\[
\tilde{W}(s) = E(e^{-sW}) = E(E(e^{-sW_i}| \text{customer arrives to state } i)) = \pi_H \tilde{W}_{HC}(s) + \pi_k \tilde{W}_{kC}(s) + \pi_L \tilde{W}_{LC}(s) + \pi_0. \quad (4.82)
\]

\[
\tilde{W}(s) = \pi_H \tilde{W}_{HC}(s) + \pi_k \tilde{W}_{kC}(s) + \pi_L \tilde{W}_{LC}(s) + \pi_0. \quad (4.84)
\]

**Lemma 4.2.1.** The proportion of time which an M/G/1 absolute priority system spends in \(H, k, L\) cycle types, or an idle state, is given by

\[
\pi_H = \frac{\rho_H(1 - \rho)}{1 - \rho_H - \rho_k}, \quad (4.85)
\]

\[
\pi_k = \frac{\rho_k(1 - \rho)}{1 - \rho_H - \rho_k}, \quad (4.86)
\]

\[
\pi_L = \frac{\rho_L}{1 - \rho_H - \rho_k}, \quad (4.87)
\]

\[
\pi_0 = 1 - \rho. \quad (4.88)
\]

**Proof.** Conway, Maxwell and Miller appeal to a result from renewal theory stating that the proportion of time spent in a state is equal to the mean duration of this state, divided by the mean delay between states [2]. This is very close to Theorem 2.5.2, which states the proportion of time which the queueing system spends in some state is
equal to the rate at which the system enters this state, multiplied by the mean duration of this state, which is our preferred approach to this proof.

Firstly, we note that each type $L$ customer entering service starts a new type $L$ cycle. Class $L$ customers arrive at rate $\lambda_L$ and the mean duration of type $L$ cycles is given in Equation (4.81); multiplying these gives

$$\pi_L = \frac{\rho_L}{1 - \rho_L - \rho_k}. \quad (4.89)$$

Type $H$ and $k$ cycles, however, are not initiated by all class $H$ and $k$ customers, but only those who arrive while the system is idle. The proportion of time during which the system is idle is $1 - \rho$, and by the PASTA property, arrivals encounter an idle system with probability $1 - \rho$. Thus the long-run rate at which class $H$ and $k$ cycles are initiated is $\lambda_H(1 - \rho)$ and $\lambda_k(1 - \rho)$, respectively. (Note that these are arrival rates in the sense of 2.5.1 — the ratio of arrival rate to time converges almost surely to this value — but such arrivals do not form a Poisson process.)

Multiplying these arrival rates by the mean durations of type $H$ and $k$ cycles from Equations (4.79) and (4.80) give

$$\pi_H = \frac{\rho_H(1 - \rho)}{1 - \rho_H - \rho_k}, \quad (4.90)$$

$$\pi_k = \frac{\rho_k(1 - \rho)}{1 - \rho_H - \rho_k}. \quad (4.91)$$

Finally, the proportion of the time the system spends idle is given by $\pi_0 = 1 - \rho$. Note that this can be determined by elementary means, as in Theorem 2.5.2, or by evaluating $1 - \pi_H - \pi_k - \pi_L$.

4.2.4 Unconditional waiting time distribution

**Theorem 4.2.2.** The waiting time for a class $k$ customer in an $M/G/1$ absolute priority queue has LST

$$\tilde{W}(s) = \frac{(1 - \rho)(\lambda_H\tilde{G}_{H+}(s) - \lambda_H - s) + \lambda_L(\tilde{G}_{LH+}(s) - 1)}{\lambda_k - \lambda_k\tilde{G}_{KH+}(s) - s}. \quad (4.92)$$

**Proof.** Equation (4.84) gives us the waiting time in terms of the different cycles.

$$\tilde{W}(s) = \pi_0 + \pi_H\tilde{W}_{HC}(s) + \pi_k\tilde{W}_{kC}(s) + \pi_L\tilde{W}_{LC}(s). \quad (4.93)$$

We substitute the proportion of time spent on each cycle type from Lemma 4.2.1 and the conditional waiting time distributions from Equations (4.75) to (4.77) to obtain
\[ \tilde{W}(s) = \pi_0 + \pi_H \tilde{W}_{HC}(s) + \pi_k \tilde{W}_{kC}(s) + \pi_L \tilde{W}_{LC}(s) \quad (4.94) \]

\[ = 1 - \rho + \frac{\lambda_H (1 - \rho)(\tilde{G}_{H+}(s) - 1) + \lambda_k (1 - \rho)(\tilde{G}_{kH^+}(s) - 1)}{\lambda_k - \lambda_k \tilde{G}_{kH^+}(s) - s} \quad (4.95) \]

\[ = \frac{(1 - \rho)(\lambda_H \tilde{G}_{H+}(s) - \lambda_H - s) + \lambda_L (\tilde{G}_{LH^+}(s) - 1)}{\lambda_k - \lambda_k \tilde{G}_{KH^+}(s) - s} \quad (4.96) \]
Chapter 5

Expected waiting time in an M/G/1 priority queue

In this chapter we will consider Kleinrock’s method for determining the expected waiting times of different customer classes [7]. We are still only considering non-preemptive systems, so the overall mean waiting times will remain independent of queueing discipline.

This method applies to a more general class of queueing disciplines, in constrast to the absolute priority discipline considered earlier.

We still have a finite set of priorities \{1 \ldots P\}, but now a customer from the priority class \(p\) has priority value \(q_p(t)\) at time \(t\), for some function \(q : \mathbb{R}_{\geq 0} \to \mathbb{R}\). Whenever the server becomes idle, it selects the queued customer with the highest priority; if multiple customers have the same priority value, the customer who arrived earliest takes priority.

The absolute priority discipline can be recovered by taking \(q_p(t) = p\), but more complicated behaviour is also possible, with customers overtaking each other in priority over time.

5.1 Waiting times

Consider a tagged customer of class \(p\) arriving to some M/G/1 priority queue. Its waiting time, if any, consists of service times of other customers. If it arrives to a busy system, some other customer will be partway through their service; we denote the remaining duration of this service by \(X_0\). After this, we may have to wait for the entire service duration of some other customers.

It is helpful to divide the customers in the second category into two groups — those who are already present in the queue when the tagged customer arrives, and those who arrive after it but are served before it. Observe that because of the priority queueing discipline, the former group may not consist of every customer who is in the queue when the tagged customer arrives — the tagged customer may have higher priority than some queued customers. Similarly, the latter group exists because customers who
arrive after the tagged customer may have a higher priority.

We denote the amount of time spent waiting for each of these groups by $W_N$ and $W_M$, respectively. Thus we can represent the total waiting time and its expectation by

$$W = X_0 + W_N + W_M$$

$$E(W) = E(X_0) + E(W_N) + E(W_M).$$

Note that some or all of the waiting times may be zero. For example, if the tagged customer arrives to an empty system all three will be zero.

The first component $X_0$ is zero for arrivals to an empty system. By Theorem 2.5.2, the proportion of arrivals who encounter a busy system with a class $i$ customer in service is $\rho_i$, and the mean residual lifetime for this service is

$$E(X_i^2) \over 2E(X_i).$$

where $X_i$ is the service time for a customer of class $i$; recall that customers of a given class have IID service times.

This can be seen by considering a modified version of this queue, where periods in which the server is processing a class $i$ customer are concatenated together. We completely ignore idle periods, and our new “time” variable measures the length of time that the queue has been busy with a class $i$ customer. Under these conditions, class $i$ service intervals form a renewal process; this modified system is always processing a single customer, and the service times of successive customers are IID.

Hence, the expected waiting time due to the customer currently in service is

$$E(X_0) = \sum_{i=1}^{P} \lambda_i E(X_i^2) E(X_i) = \sum_{i=1}^{P} \rho_i E(X_i^2).$$

Let $N_{ip}$ and $M_{ip}$ be the number of customers of class $i$ who are served before us, and who arrive either before or after us, respectively. Denoting the service time of the $j^{th}$ customer of class $i$ in each of these classes by $X_i^{(N,j)}$ and $X_i^{(M,j)}$, respectively, we have

$$W_N = \sum_{i=1}^{P} \sum_{j=1}^{N_{ip}} X_i^{(N,j)}$$

$$W_M = \sum_{i=1}^{P} \sum_{j=1}^{M_{ip}} X_i^{(M,j)}.$$
Kleinrock [7] claims that, “Since the service time for any member from group \( i \) is drawn independently from \( B_i(x) \),

\[
E(W_N) = \sum_{i=1}^{P} E(X_i)E(N_{ip}) \tag{5.7}
\]

\[
E(W_M) = \sum_{i=1}^{P} E(X_i)E(M_{ip}). \tag{5.8}
\]

We were quite skeptical of this claim, as there is dependence between the service times and number served. Consider a queue where priority increases over time, and class 1 customers accumulate priority faster than class 2 customers, and a class 1 customer arrives with two class 2 customers queued. If the first class 2 customer enters service and takes a very long time, then the class 1 customer will overtake the second class 2 customer. On the other hand, if the first class 2 customer is served very quickly, the class 1 customer will have less time to overtake second class 2 customer.

However, the result above can be shown under weaker conditions than independence between the index and summands. By Wald’s identity (Theorem 2.2.1), we only require that the events \( \{N_{ip} > n\} \) and \( \{M_{ip} > n\} \) are independent of \( \{X_{i(N,j)}|j > n\} \) and \( \{X_{i(N,j)}|j > n\} \), respectively.

The event \( \{N_{ip} > n\} \) is equivalent to the event that the \((n+1)^{th}\) customer of class \( i \) who is queued when our tagged customer arrives is served before the tagged customer.\(^1\) There can be no dependence between the time when the \((n+1)^{th}\) such customer enters service, and the service time of any customer past the \(n^{th}\). Exactly the same argument applies to the event \( \{M_{ip} > n\} \). This is equivalent to the \((n+1)^{th}\) class \( i \) customer overtaking the tagged customer in the queue, which depends only on the service times of the first \( n \) customers.

Thus, Equations (5.7) and (5.8) hold by Wald’s identity. Substituting into the overall waiting time, we have

\[
E(W_p) = E(X_0) + \sum_{i=1}^{P} E(X_i)(E(N_{ip}) + E(M_{ip})). \tag{5.9}
\]

### 5.2 Conservation laws

In a non-preemptive, stable, work conserving system, the time spent by the server on any given task does not vary with the service discipline. Moreover, the system will

\(^1\)Here we are labelling class \( i \) customers in the order in which they are served, which may not be the order in which they are in the queue; we have not ruled out non-monotonic priority functions. This does not alter the central fact that a customer’s waiting time can only depend on service times up to the point the customer enters service.
always process a customer if one is available. Thus, the queueing discipline can only change the order in which customers are processed; the total time spent on any given set of customers is unaffected by the queueing discipline. If we wish to improve the service for some class of customers, we will inevitably degrade it for some other class. We will make this notion more precise in the following section.

Recall that $U(t)$ is the unfinished work, or the amount of time it will take to service all customers who are currently in the system at time $t$. This does not include the time taken to serve any new customers who arrive during this time. Moreover, this value is independent of the order in which the customers are serviced.

At time $t$, let $N_p(t)$ be the number of customers of priority $p$ in the queue for each $p \in \{1 \ldots P\}$ and $X_p^j$ the service time required for the $j^{th}$ customer of class $p$, where $p \in \{1 \ldots P\}$ and $j \in \{1 \ldots N_p(t)\}$. Then we can divide the unfinished work into the residual service time of the current customer, plus the service times of the queued customers of each class.

\[
U(t) = X_0 + \sum_{p=1}^{P} \sum_{j=1}^{N_p(t)} X_p^j 
\]  

(5.10)

\[
E(U(t)) = E(X_0) + \sum_{p=1}^{P} E\left( \sum_{j=1}^{N_p(t)} X_p^j \right) 
\]  

(5.11)

Now the service times $X_p^j$ are independent across all $p$ and $j$, with customers of the same class $p$ having the same service time distribution function $B_p$. Moreover, we are considering the system at time $t$, while all service times are in the future, so $N_p(t)$ is independent of all $X_p^j$.

Thus by Wald’s identity

\[
E(U(t)) = E(X_0) + \sum_{p=1}^{P} E(X_p)E(N_p(t)).
\]  

(5.12)

By Little’s law (Theorem 2.5.1), $E(N_i(t))$ has a limit $\bar{N}_i$ as $t \to \infty$. Hence, $E(U(t))$ also has a limit. In fact, we have already seen in Section 3.1 that $U(t)$ is equal to the waiting time for an arrival to a first-come first-served queue at $t$, and found its limiting distribution (or rather, its Laplace-Stieltjes transform). Hence, the limit of $U(t)$ is simply $W$, the waiting time for an arrival to a FCFS queue in steady state. Taking limits in Equation (5.12) and applying Little’s Law, we find
\[ E(W) = E(X_0) + \sum_{p=1}^{P} E(X_p) \bar{N}_p \]  
\[ = E(X_0) + \sum_{p=1}^{P} E(X_p) \lambda_p E(W_p) \]  
\[ = E(X_0) + \sum_{p=1}^{P} \rho_p E(W_p). \]  

(5.13)  
(5.14)  
(5.15)

Recall Theorem (3.3.4), which gave the Pollaczek-Khinchin formula for the mean waiting time:

\[ E(W) = \frac{\lambda E(X^2)}{2(1 - \rho)}. \]  

(5.16)

It turns out that the second moment of service time is closely related to the residual lifetime. Firstly, note that the number of customers serviced, and their service times, are independent of the queueing discipline for any stable queue \((\rho < 1)\), as every arriving customer will be serviced. Moreover, because we assume that arrivals form a Poisson process, arrival times are independent of any behaviour of the queue prior to their arrival. Thus, \(E(X_0)\) is constant across all queueing disciplines; we will calculate it for a FCFS queue for convenience.

We construct a renewal process to find \(E(X_0)\) in a similar manner to the previous section. Here, we have no priority classes, so we concatenate all busy periods together and consider each customer service period to be an epoch.\(^2\) Thus, residual lifetimes in this renewal process have limiting PDF \(\frac{1-B(x)}{E(X)}\) and mean \(\frac{E(X^2)}{2E(X)}\).

This system gives the residual lifetime for an arrival to a FCFS M/G/1 queue during its busy period; note that Poisson arrivals are equally likely to occur at any point during a given renewal epoch. The probability that an arrival finds a busy system is \(\rho\). On the other hand, arrivals to an idle system experience zero residual lifetime. Hence

\[ E(X_0) = \frac{\rho E(X^2)}{2E(X)} \]  
\[ \therefore E(X^2) = \frac{2E(X_0)}{\lambda}. \]  

(5.17)  
(5.18)

Substituting this back into the Pollaczek-Khinchin formula, we find the expected waiting time in terms of the mean residual lifetime.

\(^2\)Note that if we impose another priority discipline on this system, and the service times vary with the customer class, then successive service times are not IID; the longer a service time \(X_n\), the higher the probability that \(X_{n+1}\) will represent a customer of high priority. However, if we consider only customers of a single class, the service times are IID.
\[ E(W) = \frac{E(X_0)}{1 - \rho}. \quad (5.19) \]

Combining this last equation and Equation (5.15) gives us a relationship between the mean residual service time and the expected waiting times of each class for a stable queue.

\[ \frac{E(X_0)}{1 - \rho} = E(X_0) + \sum_{i=1}^{P} \rho_i E(W_i) \quad (5.20) \]

\[ \rho E(X_0) \frac{1}{1 - \rho} = \sum_{i=1}^{P} \rho_i E(W_i). \quad (5.21) \]

Thus, the sum of mean waiting times weighted by their load factors is constant. Any attempt to reduce the expected waiting time for some class of customers must inevitably come at the expense of some other class.

### 5.3 Time-dependent priorities

In this section we consider perhaps the simplest model with time-varying priorities, where the customer’s priority at time \( t \) is a multiple of the time that they have spent waiting up to \( t \).

This model was first introduced by Kleinrock [23, 7], who referred to it as a time-dependent priority queue. More recent work by Stanford, Taylor and Ziedins [4] considers the same system, but refers to it as an accumulating priority queue. In this section, we will follow Kleinrock’s derivation of the expected waiting time in this system, while in Chapter 6 we will consider Stanford, Taylor and Ziedins’s method for determining the full waiting time distribution.

Under this model, each customer belongs to a priority class \( p \in \{1 \ldots P\} \), and each class has a rate of priority accumulation \( b_p \), with \( 0 \leq b_P \leq \ldots \leq b_1 \). Given these values, a customer who arrives at time \( \tau \) is assigned priority \( q_p(t) = (t - \tau)b_p \) at time \( t \). In other words, every customer’s priority increases linearly with time as they wait in the queue, with the rate of priority accumulation depending on their class. All arriving customers start at the tail of the queue, with zero priority. However, as they wait in the queue they may overtake customers who arrived before them, but are members of a lower priority class.

Kleinrock restricts his analysis of this system to exponential service times. It is not entirely clear what part of the analysis does not apply for general service times; it may be related to the derivation of \( E(M_{ip}) \), as discussed below.

We tag a customer of priority class \( p \) arriving to a system in steady state at time 0; their waiting time is denoted by \( W_p \). As before, \( M_{ip} \) is the number of customers of group \( i \) who arrive after time 0, but are served before our tagged customer, while \( N_{ip} \).
is the number of customers of class $i$ already in the queue at time 0, and are served before the tagged customer.

For $i \geq p$ we have $M_{ip} = 0$, because subsequent arrivals of equal or lower priority to $p$ can’t accumulate priority faster than our tagged customer.

For $i < p$, let $V_i$ be the last time at which a customer of priority $i$ with waiting time $W_i$ can arrive and overtake our tagged customer in the queue. As the class $i$ customer accumulates priority faster than the class $p$ customer, this is the time at which each the customers have the same priority.

\[ b_i(t - V_i) = b_pt \]
\[ : V_i = W_p \left( 1 - \frac{b_p}{b_i} \right). \]

Thus, $M_{ip}$ is the number of class $i$ arrivals in the interval $(0, V_i)$, for $i < p$. From here Kleinrock concludes that, as class $i$ customers arrive at a rate $\lambda_i$,

\[ E(M_{ip}) = \lambda_iE(V_i) \]
\[ = \lambda_iE(W_p) \left( 1 - \frac{b_p}{b_i} \right), \quad i < p. \]

We are not entirely convinced by this reasoning. There is clearly dependence between the arrivals in some interval $(0, t)$ and $W_p$; each arrival who overtakes the tagged customer will extend $W_p$. Thus, it is unclear whether, conditioning on $V_i$, class $i$ arrivals in $(0, V_i)$ form a Poisson process. It is possible that arrivals do not form a Poisson process, but nonetheless the mean arrival count is still $\lambda_iE(V_i)$. Alternatively, the unexplained restriction earlier that service times must also be memoryless may recover the Poisson property.

We believe that further analysis of this situation will need to consider delay cycles of the forms considered in the previous two chapters, as arriving customers delay the tagged customer, allowing for more customers to overtake the tagged customer, and so on. However, we do not attempt this here, because this approach has already been considered by Stanford, Taylor and Ziedins in [4], and is the topic of the next chapter. For now, we assume that Equation (5.25) is correct, and continue with Kleinrock’s analysis.

Now consider $N_{ip}$, the number of customers of priority $i$ who are already in the queue when our tagged customer arrives, and are also served before our customer. The case where $i \leq p$ is relatively simple, as these customers accumulate priority at least as fast as our customer and therefore cannot be overtaken. Such customers arrive at a rate $\lambda_i$ and have expected waiting time of $W_i$. By Little’s Law (Theorem 2.5.1), the mean number in the queue at $t = 0$ is given by

\[ E(N_{ip}) = \lambda_iE(W_i), \quad i \leq p. \]
Now consider a customer of priority $i > p$ who arrives at time $-s$ and is still in the queue at time 0. Let $u$ be the time when the earlier customer's priority is equal to our tagged customer (assuming that neither enter service beforehand). Note that because $i > p$, the tagged customer is accumulating priority faster and will take precedence after time $u$. Using the continuously accumulating priority we find the following:

$$b_p u = b_i(s + u),$$  \(5.27\)

$$u = \frac{sb_i}{b_p - b_i},$$  \(5.28\)

$$s + u = \frac{sb_p}{b_p - b_i}.$$  \(5.29\)

The earlier customer will be served before our tagged customer if and only if $W_i \leq u$. For it to be in the queue at time zero, $W_i > s$.

We want to calculate $E(N_{ip})$, the expected number of customers satisfying the two conditions above. From the above conditions and the arrival distribution we can obtain an integral for $E(N_{ip})$:

$$E(N_{ip}) = \int_0^\infty \lambda_i P\left(s < W_i(t) \leq \frac{tb_p}{b_p - b_i}\right) ds$$  \(5.30\)

$$= \lambda_i \int_0^\infty 1 - P(W_i \leq s)ds - \lambda_i \int_0^\infty 1 - P(W_i \leq \frac{sb_p}{b_p - b_i}) ds. \ (5.31)$$

Using a change of variable we can further find:

$$E(N_{ip}) = \lambda_i \int_0^\infty 1 - P(W_i \leq t) dt - \lambda_i \left(1 - \frac{b_i}{b_p}\right) \int_0^\infty 1 - P(W_i \leq y) dy$$  \(5.32\)

$$= \lambda_i E(W_i) - \lambda_i \left(1 - \frac{b_i}{b_p}\right) E(W_i)$$  \(5.33\)

$$= \lambda_i E(W_i) \frac{b_i}{b_p}, \ i > p. \ (5.34)$$

Substituting Equations (5.25), (5.26) and (5.34) into (5.9):

$$E(W_p) = E(X_0) + \sum_{i=1}^P E(X_i)(E(N_{ip}) + E(M_{ip}))$$  \(5.35\)

$$= E(X_0) + \sum_{i=1}^P E(X_i)\lambda_i E(W_i) + \sum_{i=p+1}^P E(X_i)\lambda_i E(W_i) \frac{b_i}{b_p} + \sum_{i=1}^{p-1} E(X_i)\lambda_i E(W_p) \left(1 - \frac{b_p}{b_i}\right). \ (5.36)$$
We collect the latter multiples of $E(W_p)$ and simplify the result:

$$E(W_p) = \frac{E(X_0) + \sum_{i=1}^{p} \rho_i E(W_i) + \sum_{i=p+1}^{p} \rho_i E(W_i) \frac{b_i}{b_p}}{1 - \sum_{i=1}^{p-1} \rho_i \left(1 - \frac{b_i}{b_p}\right)}.$$  \hspace{1cm} (5.37)

Rearranging Equation (5.21), the conservation law from the previous section:

$$\sum_{i=1}^{p} \rho_i E(W_i) = \frac{\rho E(X_0)}{1 - \rho} - \sum_{i=p+1}^{p} \rho_i E(W_i).$$  \hspace{1cm} (5.38)

We substitute this into our mean waiting time to arrive at its most tractable form.

$$E(W_p) = \frac{E(X_0) + \sum_{i=p+1}^{p} \rho_i E(W_i) \left(1 - \frac{b_i}{b_p}\right)}{1 - \sum_{i=1}^{p-1} \rho_i \left(1 - \frac{b_i}{b_p}\right)}.$$  \hspace{1cm} (5.39)

Observe that while the expected waiting time for priority class $p$ depends on the expected waiting times of other priority classes, only those of lower priority are required. Thus we can iteratively determine the expected waiting time of each class, starting from $P$ (the lowest priority).

The only dependence on the priority accumulation rates $b_i$ is in terms of ratios between the different rates. By manipulating these ratios we may determine the ratios between different expected waiting times.

Note however, that the net throughput of the system depends only on the arrival and service distributions, and is unaffected by the queueing discipline. We are only able to control the relative waiting times of the different priority classes.
Chapter 6

The waiting time distributions in an accumulating priority M/G/1 queue

We now consider an M/G/1 queue with accumulating priorities. Such a queue was originally proposed by Kleinrock [23, 7], who determined the mean waiting time in the M/M/1 case, which we considered in Chapter 5. Current work by Stanford, Taylor and Ziedins [4] gives the stationary distribution of waiting time. We will follow the latter, noting that it has not yet been published, and its final revision will be modified from the version we have used.

This system has $P$ priority classes; customers of each class arrive as a Poisson process with parameter $\lambda_k$. Each customer class has its own service distribution $B_k$; we require that $B_k$ has a Laplace-Stieltjes transform $\tilde{B}_k(s)$ defined in some neighbourhood of the origin. The priority of a customer of class $k$ increases linearly at rate $b_k$ while the customer is queued; thus if a customer of class $k$ arrives at time $t_0$ and is still waiting at time $t$, their priority is $b_k(t - t_0)$. We order the classes such that $b_1 > \ldots > b_P > 0$.

Kleinrock also considers a generalisation of this system, where the priority is $b_k(t - t_0)^r$ for some $r \in \mathbb{R}_{\geq 0}$ [23, 7]. We will restrict our attention to the case described above, where $r = 1$.

We define the total arrival rate to be $\lambda = \sum_{p=1}^{P} \lambda_p$ and the overall service time distribution is

$$B(t) = \sum_{k=1}^{P} \frac{\lambda_k}{\lambda} B_k(t) \quad (6.1)$$

with Laplace-Stieltjes transform

$$\tilde{B}(s) = \sum_{k=1}^{P} \frac{\lambda_k}{\lambda} \tilde{B}_k(s). \quad (6.2)$$

For convenience we denote the mean waiting time of each class $k$ by $\mu_k^{-1}$ and overall mean by $\mu^{-1}$, where
With a single class, the accumulating priority queue is simply a FCFS queue, while in the limit $b_i/b_{i+1} \to \infty$ it becomes an absolute priority queue, as new customers will overtake queued customers of lower priority class almost instantly after their arrival.

We can consider the virtual work process for this class in the same manner as in Chapter 3. Because this queue is simply an M/G/1 queue with an added priority discipline, the virtual work process is exactly the same as the FCFS version. This will give us the overall mean waiting time, for example. However, if we want to account for the different priority classes, we need to consider a new, related process, which we describe below.

### 6.1 Maximum priority process

The maximum priority process was introduced by Stanford, Taylor and Ziedins [4] in order to analyse the accumulating priority queue. We largely follow their work, although the upper bound in Theorem 6.1.1 and partial results regarding the stationary distribution is original. The maximum priority process can be viewed as an extension of the maximum waiting time process described in Subsection 3.1.2.

At time $t$, the process $M(t) = (M_1(t), \ldots, M_P(t))$ is a vector of length $P$ giving the maximum possible priority which could have been accumulated by a queued customer of each class, assuming that we can observe only the server. That is, we can observe any customer who is now or has previously been in service, but not currently queued customers.

Clearly, if the server is empty, then the maximum priority process must be the zero vector. If a customer arrives to an empty system and commences service at time $t_0$, then another customer of any class $k$ may arrive arbitrarily soon after $t_0$ and begin accumulating priority at rate $b_k$. Thus, each $M_k$ increases at a constant rate $b_k$.

If a customer finishes service and another customer $C'$ enters service at time $t_1$, all queued customers must have priority less than that of $C'$. Thus $M_i(t_1)$ will be reduced to the priority of $C'$ upon entering service at time $t_1$. Lower priority components $M_k(t_1)$ will be reduced to the priority of $C'$ if they were higher just before time $t$; otherwise they will continue increasing at rate $b_k$.

To formally specify the behaviour of the maximum priority process, let $T_n$ be the time at which the $n^{th}$ customer arrives, and $D_n$ be the time at which the same customer departs. (Note that the $T_n$ must be ordered such that $T_1 < T_2 < \ldots$, but the priority discipline may cause the $D_n$ to be reordered as some customers overtake others.) Fur-
thermore, let $V_n(t)$ be the priority of the $n$th arrival at time $t$. If the customer has not arrived yet, we define their priority to be zero.

Then at all departure instants $s$,

\begin{align*}
M_1(s) &= \max\{V_n(s^-)|T_n < s, D_n > s\} \quad (6.5) \\
M_{k+1}(s) &= \min\{M_{k+1}(s^-), M_k(s)\} \quad k \in \{2 \ldots P\}. \quad (6.6)
\end{align*}

At all other times $t$ and all priority classes $k \in \{1 \ldots P\}$,

\begin{align*}
M_k(t) &= 0 \text{ if the system is idle;} \quad (6.7) \\
M'_k(t) &= b_k \text{ if the system is busy.} \quad (6.8)
\end{align*}

**Theorem 6.1.1.** For any $1 \leq k < l \leq P$ and any time $t$ at which the system is busy,

\begin{equation}
1 \leq \frac{M_k(t)}{M_l(t)} \leq \frac{b_k}{b_l}. \quad (6.9)
\end{equation}

**Proof.** When a customer arrives to an empty system, $M_k$ and $M_l$ start increasing at constant rates $b_k$ and $b_l$, respectively; note that $b_k > b_l$.

If a customer leaves the system at time $t$ and the next customer to be served has priority in $(M_l(t^-), M_k(t^-))$, then $M_k(t)$ is reduced to the new customer’s accumulated priority, but $M_l$ is unaffected (but is still lower than $M_k(t)$). Thus, the ratio of $M_k$ to $M_l$ is reduced, but to no less than 1.

If the new customer has priority lower than $M_l(t^-)$ as well, then both $M_k$ and $M_l$ are reduced to the new value. From here both processes continue to increase at rates $b_k$ and $b_l$. Thus their ratio increases from 1 and will approach $b_k/b_l$ as the new customer’s time in service increases, but can never reach that level until a new busy period starts.

The ratio thus continues increasing towards $b_k/b_l$ and dropping no lower than 1 until the system becomes idle, at which point both processes drop to zero. The theorem then follows.

We believe that the maximum priority process must have a stationary distribution for a stable queue, as it is regenerative; intervals between the end of successive busy periods are independent of each other. We will not attempt to prove this, however, and simply assume that it exists for the purposes of the following paragraphs. (The main results of this chapter do not rely on stationary properties of the maximum priority process.)

Firstly, note that the stationary distribution has a point mass of $1 - \rho$ at the origin, as the maximum priority process is the zero vector whenever the system is idle, and nonzero with probability 1 when the system is busy.
Moreover, the maximum priority process increases along the straight line $M_1(t)/M_k(t) = b_1/b_k$ in $\mathbb{R}^P$ for all $k \in \{2 \ldots P\}$ during the first service in a busy period. Such services begin at rate $\lambda(1 - \rho)$ and have a mean length of $1/\mu$, so the system spends a fraction $\rho(1 - \rho)$ of the time on this line. Thus, this line in $P$-dimensional space has a nonzero mass in the stationary distribution of $M$. Along this line, the mass is distributed according to the overall service distribution $B$.

The remaining $\rho^2$ fraction of the time is spent on the second or later arrival to a busy period. In these periods, the maximum priority process may take values anywhere in the region specified by Theorem 6.1.1.

If we consider the case where the first $n$ customers in a busy period are served with priority above $M_j$, we see that $M_j(t)/M_k(t) = b_j/b_k$ for all $k > j$, but the first $j - 1$ components of $M$ will not obey this ratio. Thus, for any $j = 2 \ldots P - 1$, there is a $j$-dimensional region in $\mathbb{R}^P$ with positive probability. This probability is part of the $\rho^2$ above; we will not attempt to calculate its exact value.

**Theorem 6.1.2.** Given $M_k(t)$, the priorities of queued customers of class $k$ form a non-homogeneous Poisson process with rate $\lambda_k/b_k$ on the interval $[0, M_k(t))$ and zero elsewhere.

**Proof.** The maximum priority process $M_k(t)$ effectively tells us that the earliest possible time at which a class $k$ customer could have arrived is $M_k(t)/b_k$. The construction of the process, combined with the memoryless property of Poisson arrivals, means that the process gives no information about arrivals more recently than $M_k(t)/b_k$. Moreover, the maximum priority process for any other class $j \neq k$ gives us no more information than simply $M_k(t)$ on its own.

Thus, given $M(t)$, the arrival times of queued class $k$ customers occur as a non-homogeneous Poisson process with rate $\lambda_k/b_k$ over the interval $(t - M_k(t)/b_k, t)$ and zero elsewhere. Their accumulated priorities at time $t$ are a dilation of the time since their arrival. Thus the accumulated priorities of class $k$ customers occur according to a Poisson process with rate $\lambda_k/b_k$ on the interval $(0, M_k(t))$ and zero elsewhere. $\square$

**Corollary 6.1.3.** Given $M(t)$, the accumulated priority of all queued customers at time $t$ is distributed as a non-homogeneous Poisson process with rate

$$r(v) = \begin{cases} 0, & v > M_1(t) \\ \frac{\sum_{p=1}^{k} \lambda_p}{b_p}, & M_{k+1}(t) < v \leq M_k(t). \end{cases} \quad (6.10)$$

**Proof.** For each interval of the form $[M_{k+1}(t), M_k(t))$, superimpose each Poisson process given in Theorem 6.1.2 above. For notational convenience, we set $M_{P+1}(t) = 0$ for all $t$. $\square$
6.2 Accreditation intervals

Accreditation intervals are a type of delay cycle, defined in terms of the maximum priority process. They allow us to use the delay cycle approach of Conway, Maxwell and Miller [2] to analyse the accumulating priority queue, in a similar manner to the FCFS queue in Chapter 3 and the absolute priority queue in Chapter 4.

We say that a customer \( C \) is accredited relative to class \( k \) at time \( t \) if their accumulated priority is greater than \( M_k(t) \). This can only happen if \( C \) is of some class \( j \), where \( j < k \), as no class \( k \) customer can have more priority at time \( t \) than \( M_k(t) \), and \( M_l(t) \leq M_k(t) \) for all lower priorities \( l > k \). A customer who becomes accredited relative to class \( k \) will retain this accreditation until they enter service, as \( M_k(t) \) increases at a rate \( b_k \), while the priority of \( C \) increases linearly at some rate higher than \( b_k \).

An class \( k \) accreditation interval, or accredited busy period, is a period in which the system is busy, and every customer to enter service is accredited with respect to class \( k \). Note that \( M_k \) is increasing at rate \( b_k \) throughout this period; it will only decrease if some customer is served at time \( t \) with lower accumulated priority than \( M_k(t) \), meaning that the accreditation interval has ended. Thus, the priority required to become accredited increases at a constant rate \( b_k \) throughout the accreditation interval.

The accreditation interval begins at some time \( s \) when a customer with accumulated priority \( v \) commences service. If the server was previously idle, then \( v = 0 \); otherwise \( v \) will be strictly positive with probability 1.

**Lemma 6.2.1.** While the system is busy, customers of class \( j \leq k \) become accredited relative to class \( k + 1 \) as a Poisson process with rate \( \lambda_j \frac{b_j - b_{k+1}}{b_j} \).

Customers of any class become accredited relative to class \( k + 1 \) as a Poisson process with rate

\[
\Lambda_{1,k} = \sum_{j=1}^{k} \lambda_j \frac{b_j - b_{k+1}}{b_j}. \tag{6.11}
\]

**Proof.** The arrival of customers forms a Poisson process. Assume that some customer of class \( j \) arrives at \( t_1 \) and becomes accredited at \( s_1 \), and a second class \( j \) customer arrives at \( t_2 \). (Note that we do not specify any relationship between \( s_1 \) and \( t_2 \).) Then the first customer must have priority \( (s_1 - t_1)/b_j \) when it is accredited at time \( s_1 \). Recall that the accumulated priority required for accreditation increases at rate \( b_k \), and so if the second customer is accredited at time \( s_2 \), it must have priority \( (s_1 - t_1)b_j + (s_2 - s_1)b_k \). We then see that

\[
(s_2 - t_2)b_j = (s_2 - s_1)b_k + (s_1 - t_1)b_j \tag{6.12}
\]

which gives

\[
s_2 - s_1 = \frac{b_j}{b_j - b_k}(t_2 - t_1). \tag{6.13}
\]
Thus, the intervals between successive accreditations are simply dilated versions of the intervals between arrivals. Hence class \( j \) customer accreditations relative to class \( k + 1 \) form a Poisson process with parameter \( \lambda_j \frac{b_j - b_{k+1}}{b_j} \). Summing over all such classes gives \( \Lambda_{1,k} \) above.

We say that a customer is \textit{served at priority level} \( k \) if they are served at time \( t \) with a priority in the range \( (M_{k+1}(t), M_k(t)) \); in other words, they must be guaranteed to take precedence over all customers of class \( j > k \).

**Corollary 6.2.2.** The proportion of all class \( j \) customers arriving during a busy period who are served at priority level \( k \) is \( \frac{b_k - b_{k+1}}{b_j} \).

**Proof.** During a busy period, customers of class \( j \leq k \) become accredited relative to class \( k + 1 \) at rate \( \frac{b_j - b_{k+1}}{b_j} \lambda_j \).

If \( j = k \), then these customers cannot become accredited relative to class \( k \). If \( j < k \), then class \( j \) customers may also become accredited relative to class \( k \). During a busy period, such accreditations occur according to a Poisson process of rate \( \frac{b_j - b_{k+1}}{b_j} \lambda_j \). Thus, some class \( j \) customers become accredited relative to class \( k + 1 \) but not class \( k \), and hence they must enter service at time \( t \) with a priority in the range \( [M_{k+1}(t), M_k(t)] \).

Combining these results, for any \( j \leq k \), the proportion of class \( j \) customers who are accredited relative to class \( k + 1 \) but not class \( k \) when they enter service is given by

\[
\frac{b_k - b_{k+1}}{b_j}.
\]

(6.14)

**Corollary 6.2.3.** The waiting time of a customer served during a busy period at priority level \( k \) is distributed as

\[
\tilde{\beta}_k(s) = \frac{1}{\Lambda_k} \sum_{j=1}^{k} \frac{b_k - b_{k+1}}{b_j} \lambda_j \tilde{B}_j(s)
\]

(6.15)

where

\[
\Lambda_k = \sum_{j=1}^{k} \frac{b_k - b_{k+1}}{b_j} \lambda_j.
\]

(6.16)

**Proof.** The waiting time of a customer of class \( j \) served at priority level \( k \) has distribution \( B_j \), as service times are independent of anything which happens prior to the customer entering service. The proportion of class \( j \) customers who are served at priority level \( k \geq j \) is given above as \( \frac{b_k - b_{k+1}}{b_j} \), and the total rate at which class \( j \) arrivals is \( \lambda_j \). Taking the weighted sum over all such \( j \) using the rates above gives us the overall waiting time, \( \tilde{\beta}_k \).

\[
\square
\]
Lemma 6.2.4. A class $k$ accreditation interval commenced by a customer with service distribution $B_0$ has duration distributed as

$$\tilde{D}^{(k)}_{B_0}(s) = B_0(s + \Lambda_{1,k} - \Lambda_{1,k} \tilde{G}_{B^+}(s))$$

(6.17)

where

$$\tilde{\beta}^{(k)}_{1,k}(s) = \sum_{i=1}^{k} \frac{\lambda_i (b_i - b_{k+1})}{\Lambda_{1,k} b_i} \tilde{B}_i(s)$$

(6.18)

and $\tilde{G}_{B^+}$ is the LST of a busy period of customers accredited with respect to class $k+1$, which satisfies

$$\tilde{G}_{B^+}(s) = \tilde{\beta}^{(k)}_{1,k}(s + \Lambda_{1,k} - \Lambda_{1,k} \tilde{G}_{B^+}(s)).$$

(6.19)

Proof. Customers become accredited relative to class $k+1$ as a Poisson process with rate $\Lambda_{1,k}$ during a busy period, by Lemma 6.2.1. While customers accredited relative to class $k+1$ are queued, no other customer will be served. Thus a class $k$ accreditation interval is a delay cycle. We have specified that the initial delay has distribution function $B_0$, and blocking tasks are customers accredited relative to class $k+1$.

If $j > k$, no customer of class $j$ can be accredited relative to class $k$. By Corollary 6.2.2, the fraction of customers of class $j \leq k$ arriving during a busy period who become accredited with respect to class $k+1$ is

$$\frac{b_j - b_{k+1}}{b_j}.$$ 

(6.20)

Thus, class $j$ contributes a fraction $\frac{\lambda_i (b_i - b_{k+1})}{\Lambda_{1,k} b_i}$ of those customers served while accredited relative to class $k+1$; summing over all $j \leq k$ gives us $\tilde{\beta}^{(k)}_{1,k}$ above.

Substituting the initial delay duration, arrival rate of blocking tasks and blocking task duration into Theorem 4.1.1 gives the delay cycle duration $\tilde{D}^{(k)}_{B_0}(s)$. 

Lemma 6.2.5. A class $k$ accreditation interval initiated by a customer with priority distribution function $B_0$ can be partitioned into class $k-1$ accreditation intervals. The first class $k-1$ accreditation interval has length distributed as $D^{(k-1)}_{B_0}(s)$. Later class $k-1$ accreditation intervals are initiated at rate $\Lambda_{k}B_0(s)$ and have length distributed as $\tilde{D}^{(k-1)}_{\tilde{\beta}_k}(s)$.

Proof. The first class $k-1$ accreditation interval begins with the first job of the overall interval. Later class $k-1$ accreditation intervals must be started by a customer who is accredited relative to class $k+1$ (otherwise the overall class $k$ accreditation interval would end) but not class $k$ (otherwise the previous class $k-1$ accreditation interval
would not have finished). Thus, Lemma 6.2.4 immediately shows that the first class $k - 1$ accreditation interval has length distributed as $D_{B_0}^{(k-1)}(s)$.

Later intervals are initiated by customers served at level $k$; by Corollary 6.2.2 such customers arrive during a busy period at rate $\Lambda_k$ and have overall distribution

$$\beta_k(s) = \frac{1}{\Lambda_k} \sum_{j=1}^{k} \lambda_j \frac{b_k - b_{k+1}}{b_j} B_j(s).$$

(6.21)

Hence all class $k - 1$ accreditation intervals beyond the first have length distributed as $D_{\beta_k}^{(k-1)}(s)$. \hfill $\square$

Note that this holds for $k = 1$, as well. No customer can be accredited with respect to class 1, so a “class 0” accreditation interval has arrival rate zero. Substituting $k = 0$ into the formula for $\Lambda_k$ gives this result. Thus, this division of a class 1 accreditation interval is equivalent to considering each class 1 customer to be served at level 1 as a separate delay cycle.

Theorem 6.2.6. The proportion of time spent on customers served at priority level $l$ or higher is

$$\sigma_l = \sum_{j=1}^{l} \rho_j \frac{(b_j - b_{l+1})}{b_j}.$$  

(6.22)

Proof. Customers of class $j \leq l$ become accredited with respect to class $l + 1$ at rate $\lambda_j(b_j - b_{l+1})/b_j$, by Lemma 6.2.1. Given that service times are independent of all waiting times prior to the service, the proportion of time spent on such customers must be $\rho_j(b_j - b_{l+1})/b_j$.

Summing over $j$, we find the above expression for the proportion of time spent on customers who are accredited relative to class $l + 1$, or equivalently, customers served at priority level $l$ or higher. \hfill $\square$

### 6.3 Priority distribution for an accredited customer

The following theorem is adapted from Stanford, Taylor and Ziedins [4], who have in turn based their argument on Conway, Maxwell and Miller [2], which we examined in Chapter 4. Stanford et al prove a simpler version of this theorem, with only two classes [4]. Then, they apply it to a system with multiple classes, replacing individual jobs with class $k - 1$ accreditation intervals, and replacing the arrival rate of a single high priority class with an $\sum_{i=1}^{k} \lambda_i b_i / b_k$.

We believe that this is correct, but it is unclear why class $k - 1$ accreditation intervals would start at that rate, at least in the most recent revision we have seen (10
September). We attempt to generalise the theorem to a system with multiple classes here. Note that the expression for $\tilde{U}_{B_0}^{(k)}(s)$ below is the same as that obtained in [4] by substituting the cumulative rates and delay cycle durations in Lemmas 8.1 and 8.2 into the priority distribution in Theorem 6.1 (note that these three references are to [4]).

**Theorem 6.3.1.** Consider a class $k$ accreditation interval which starts at time $t_0$ with a customer who has accumulated priority $v$ and whose length has mean $\mu_0^{-1}$ and distribution $B_0$. Also, assume we have tagged some class $k$ customer who is served during this interval. The accumulated priority of the tagged customer beyond $v$ at the time their service commences has LST

$$\tilde{U}_{B_0}^{(k)}(s) = \frac{\mu_0(1 - \frac{1}{\nu} \sum_{i=1}^{k} \lambda_i \frac{b_i}{b_k} b_{k+1}(b_{k+1}) \tilde{D}_{B_0}^{(k)}(b_{k+1} s) - \tilde{D}_{B_0}^{(k-1)}(b_k s))}{(1 - \frac{b_k}{b_{k+1}})(b_k s + \sum_{l=1}^{k} \lambda_l \frac{b_l}{b_k} (-1 + \tilde{D}_{B_0}^{(k-1)}(b_k s)))}.$$  \hspace{1cm} (6.23)

where $D_{B}^{(l)}$ is the distribution function of a class $l$ accreditation interval initiated by a job with distribution $B$, and $\nu^{-1}$ is the expected length of a class $k - 1$ accreditation interval initiated by a job served at level $k$.

**Proof.** We divide the class $k$ accreditation interval into sub-intervals, as in Chapters 3 and 4. The initial sub-interval consists of the initial delay cycle; denote its duration by $S_0$. Sub-interval $j + 1$ has duration $S_{j+1}$, and consists of the service time for each class $k - 1$ accreditation interval initiated by a customer who becomes accredited in $(t_0 + \sum_{i=0}^{j-1} S_i, t_0 + \sum_{i=0}^{j} S_i]$. We denote the total time from the beginning of the accreditation interval until the end of interval $j$ by $\delta_j$; thus

$$\delta_j = \sum_{i=0}^{j} S_j. \hspace{1cm} (6.24)$$

We denote the LST of $S_j$ by $\tilde{H}_j(s)$, and find a recursive relation for the $\tilde{H}_j(s)$ by conditioning on the previous interval. In this case, we need to consider the number of arrivals of each class who become accredited with respect to class $k + 1$ in interval $j - 1$. Each such arrival induces a class $k - 1$ accredited busy period whose length is distributed as $D_{B_{\beta_k}}^{(k-1)}$.

$$\tilde{H}_j(s) = E(e^{-sS_j})$$

$$= E(E(e^{-sS_j}|S_{j-1} = t, N_1 = n_1, \ldots, N_k = n_k))$$

$$= E\left( E\left( \prod_{l=1}^{k} \tilde{D}_{B_{\beta_k}}^{(k-1)}(s)_l | S_{j-1} = t, N_1 = n_1, \ldots, N_k = n_k \right) \right).$$  \hspace{1cm} (6.27)
Now consider a customer of class \( l \leq k \) who becomes accredited with respect to class \( k + 1 \) at time \( t_a \), in interval \( j \). Equating the priority of our class \( l \) customer and the priority required for accreditation, we find

\[
\begin{align*}
\quad u + (t_a - t_v) b_l &= u + (t_a - t_0) b_{k+1} \\
\therefore t_v &= t_a - t_0 b_l + t_0 \frac{b_{k+1}}{b_l}.
\end{align*}
\] (6.28)

Now \( t_a \in (t_0 + \delta_{j-2}, t_0 + \delta_{j-1}] \), and our class \( l \) customer becomes accredited in interval \( j - 1 \). Hence

\[
\begin{align*}
\quad t_v &\in \left( t_0 + \frac{b_l - b_{k+1}}{b_l} \delta_{j-2}, t_0 + \frac{b_l - b_{k+1}}{b_l} \delta_{j-1} \right].
\end{align*}
\] (6.29)

Thus, each customer of class \( l \) must achieve priority \( v \) in an interval of width \( \frac{b_l - b_{k+1}}{b_l} S_{j-1} \), and hence must also arrive to the system in an interval of the same width. The arrival of customers of each class \( l \) forms a Poisson process of rate \( \lambda_l \) independent of all other classes.

This means that each \( N_l \) has a Poisson distribution with parameter \( \lambda_l \frac{b_l - b_{k+1}}{b_l} t \). From here, some computation gives us our recursive relation.

\[
\begin{align*}
\quad \widetilde{H}_j(s) &= EE \left( \prod_{l=1}^{k} \sum_{n_l=0}^{\infty} \frac{(\lambda_l \widetilde{D}_{\beta_k}^{(k-1)}(s) (b_l - b_{k+1}) t)^{n_l}}{b_l^{n_l} n_l!} \right) | S_{j-1} = t \right) \\
&= EE \left( \prod_{l=1}^{k} \exp(-\lambda_l t \frac{b_l - b_{k+1}}{b_{k+1}} (1 - \widetilde{D}_{\beta_k}^{(k-1)}(s))) | S_{j-1} = t \right) \\
&= \int_{0}^{\infty} \exp(-\sum_{l=1}^{k} \lambda_l t \frac{b_l - b_{k+1}}{b_{k+1}} (1 - \widetilde{D}_{\beta_k}^{(k-1)}(s))) dH_{j-1}(t) \\
&= \widetilde{H}_{j-1} \left( \sum_{l=1}^{k} \lambda_l \left( 1 - \frac{b_l}{b_{k+1}} \right) (1 - \widetilde{D}_{\beta_k}^{(k-1)}(s)) \right).
\end{align*}
\] (6.31)

Now we consider the waiting time of a tagged class \( k \) customer who becomes accredited relative to class \( k + 1 \) in interval \( j \), after it attains priority \( v \). Let \( A_j \) be the event that the tagged customer becomes accredited with respect to class \( k + 1 \) during interval \( j \). Also, we define \( Y \) to be the remaining time in the interval in which our tagged customer achieved priority \( v \).

We condition on the length of interval \( j \), as well as \( Y \) and the number \( N_l \) of arrivals of each class \( l \leq k \) who are accredited prior to our marked customer in interval \( S_j \). Given these parameters, the waiting time is \( Y \), plus the time from the end of this arrival interval until the end of the accreditation interval, plus the time to serve any
Customers who become accredited in interval $j$ attain priority $v$ in this interval.

Figure 6.1: Determining the waiting time distribution of an accumulating priority queue.
customers (of any class) who arrived in the $j^{th}$ arrival interval and become accredited prior to our tagged customer.

Figure 6.1 shows an example of this situation. In interval $j$, class $k$ customers $C_1$ and $C_2$ become accredited with respect to class $k + 1$. We can trace their accreditation times back to an earlier interval in which they attained priority $v$. The waiting time for $C_2$ consists of three parts: $Y$, the residual length of this interval; then the time delay between the end of the priority $v$ interval and the end of interval $j$; and then the time to process the class $k − 1$ accreditation interval induced by $C_1$ (including $C_1$ itself, along with any customers who become accredited relative to class $k$ while $C_1$ is in service).

\[
E(e^{-Sv/b_k}|A_j, S_j = t, Y = y, N_1 = n_1 \ldots N_k = n_k) = e^{-sy} \exp\left(-s \frac{b_{k+1}}{b_k} t \right) E\left( \exp\left(-s \frac{b_{k+1}}{b_k} \delta_{j-1} \right) \right) \prod_{l=1}^{k} \bar{D}_{\beta_k}^{(k-1)}(s)^{n_l}. \tag{6.36}
\]

Now each $N_l$ has a Poisson distribution, as class $l$ customers form a Poisson process. If a customer becomes accredited in interval $S_j$, they must arrive in an interval of width $t(b_l - b_{k+1})/b_l$, as determined before. However, we are only interested in those customers who become accredited prior to our tagged customer, who arrives with $y$ time left in its interval, which is scaled down from $S_j$ by a factor $b_{k+1}/b_k$. Thus, the width of the interval in which a class $l$ customer may become accredited is reduced by an amount $yb_k/b_l$, and so each $N_l$ has a Poisson distribution with parameter $\lambda_l((1 - b_{k+1}/b_l)t - b_k y/b_l)$.

\[
E(e^{-Sv/b_k}|A_j, S_j = t, Y = y) = e^{-sy} \exp\left(-s \frac{b_{k+1}}{b_k} t \right) E\left( \exp\left(-s \frac{b_{k+1}}{b_k} \delta_{j-1} \right) \right) \times \prod_{l=1}^{k} \sum_{n_l=0}^{\infty} \exp\left(-\lambda_l\left(1 - \frac{b_{k+1}}{b_k} t - \frac{b_k}{b_l} y\right)\right) \left(\frac{\lambda_l \bar{D}_{\beta_k}^{(k-1)}(s)(b_l - b_{k+1})t - b_k y}{b_l^{n_l} n_l!}\right). \tag{6.37}
\]

\[
= e^{-sy} \exp\left(-s \frac{b_{k+1}}{b_k} t \right) E\left( \exp\left(-s \frac{b_{k+1}}{b_k} \delta_{j-1} \right) \right) \times \prod_{l=1}^{k} \exp\left[-\lambda_l\left(\frac{(b_l - b_{k+1})t - b_k y}{b_l} (1 - \bar{D}_{\beta_k}^{(k-1)}(s))\right)\right]. \tag{6.38}
\]

Now $Y$ is the residual lifetime of the interval in which class $k$ customers attain priority $v$. This is a very similar situation to the cases in Chapters 3 and 4, but now $Y$ measures the residual lifetime of a shifted and rescaled version of interval $j$, as shown in Figure 6.1. The scaling factor is $(b_k - b_{k+1})/b_k$, so now $Y$ and $S_j$ have limiting joint density
\[
\frac{b_k}{(b_k - b_{k+1})E(S_j)} dydH_j(t), \quad 0 \leq y \leq \frac{b_k - b_{k+1}}{b_k} t.
\]  

(6.40)

Using this joint density, we can integrate out the conditioning on \(Y\) and \(S_j\).

\[
E(e^{sV/b_k} | A_j) = \int_0^\infty \int_0^{t \frac{b_k - b_{k+1}}{b_k}} \exp \left( -s \frac{b_{k+1}}{b_k} t \right) \exp \left[ \sum_{l=1}^{k} \lambda_l \frac{b_l - b_{k+1}}{b_l} t(-1 + \tilde{D}_{\beta_k}^{(k-1)}(s)) \right] \times 
\]

\[
\exp \left( -\sum_{l=1}^{k} \lambda_l \frac{b_k}{b_l} y(-1 + \tilde{D}_{\beta_k}^{(k-1)}(s)) \right) \times 
\]

\[
E \left( \exp \left( -s \frac{b_{k+1}}{b_k} \delta_{j-1} \right) \right) \frac{b_k}{(b_k - b_{k+1})E(S_j)} dydH_j(t).
\]  

(6.41)

We evaluate the inner integral with respect to \(y\).

\[
E(e^{sV/b_k} | A_j) = \int_0^\infty b_k E \left( \exp \left( -s \frac{b_{k+1}}{b_k} \delta_{j-1} \right) \right) \exp \left( -s \frac{b_{k+1}}{b_k} t \right) \times 
\]

\[
\exp \left( \sum_{l=1}^{k} \lambda_l \frac{b_l - b_{k+1}}{b_l} t(-1 + \tilde{D}_{\beta_k}^{(k-1)}(s)) \right) \times 
\]

\[
1 - \exp \left( -st \frac{b_k - b_{k+1}}{b_k} - \sum_{l=1}^{k} \lambda_l \frac{b_k - b_{k+1}}{b_l} t(-1 + \tilde{D}_{\beta_k}^{(k-1)}(s)) \right) 
\]

\[
\frac{1}{s + \sum_{l=1}^{k} \lambda_l \frac{b_k}{b_l} (-1 + \tilde{D}_{\beta_k}^{(k-1)}(s))} dH_j(t).
\]  

(6.42)

\[
(6.43)
\]

We evaluate the remaining integral using the recursive relationship from Equation (6.34).

Next, we multiply each term in the fraction to get a difference of two exponentials in \(t\) as the integrand.

\[
E(e^{sV/b_k} | A_j) = \int_0^\infty \frac{E \left( \exp \left( -s \frac{b_{k+1}}{b_k} \delta_{j-1} \right) \right)}{s + \sum_{l=1}^{k} \lambda_l \frac{b_k}{b_l} (-1 + \tilde{D}_{\beta_k}^{(k-1)}(s))} \frac{b_k}{(b_k - b_{k+1})E(S_j)} \times 
\]

\[
\left[ \exp \left( -\frac{b_{k+1}}{b_k} st + \sum_{l=1}^{k} \lambda_l \frac{b_l - b_{k+1}}{b_l} t(-1 + \tilde{D}_{\beta_k}^{(k-1)}(s)) \right) - \exp(-st) \right] dH_j(t).
\]  

(6.44)

We evaluate the remaining integral using the recursive relationship from Equation (6.34).
\[ E(e^{sV/b_k}|A_j) = \frac{E(\exp(-s\frac{b_{k+1}}{b_k}\delta_{j-1}))}{s + \sum_{i=1}^{k} \lambda_i \frac{b_k}{b_i} (-1 + \tilde{D}_{\beta_k}^{(k-1)}(s)) (b_k - b_{k+1})E(S_j)} \times \]
\[ \left[ E\left( \exp\left( -\frac{b_{k+1}}{b_k} sS_j + sS_{j+1} \right) \right) - E(e^{-sS_j}) \right] \] (6.45)
\[ b_k (E(\exp(-s(S_{j+1} + \frac{b_{k+1}}{b_k}\delta_{j}))) - E(\exp(-s(S_j + \frac{b_{k+1}}{b_k}\delta_{j-1})))) = (b_k - b_{k+1})E(S_j)(s + \sum_{i=1}^{k} \lambda_i \frac{b_k}{b_i} (-1 + \tilde{D}_{\beta_k}^{(k-1)}(s))) \] (6.46)

Our arrivals are Poisson, so the probability of arriving to a given system state is equal to the long-run proportion of time which the system spends in that state. Hence

\[ P(A_j) = \frac{E(S_j)}{E(S)} \] (6.47)
\[ \therefore E(e^{sV/b_k}) = \sum_{j=0}^{\infty} b_k (E(\exp(-s(S_{j+1} + \frac{b_{k+1}}{b_k}\delta_{j}))) - E(\exp(-s(S_j + \frac{b_{k+1}}{b_k}\delta_{j-1})))) \]
\[ = \frac{b_k (E(\exp(-s\frac{b_{k+1}}{b_k}S))) - E(\exp(-sS_0)))}{(b_k - b_{k+1})E(S)(s + \sum_{i=1}^{k} \lambda_i \frac{b_k}{b_i} (-1 + \tilde{D}_{\beta_k}^{(k-1)}(s)))}. \] (6.48)

Now \( S \) is the length of the entire accredited busy period; by Lemma 6.2.4 it has LST \( \tilde{D}_{B_0}^{(k)}(s) \).

We could use Theorem 4.1.2 to find \( E(S) \). However, in order to ensure that our results are of the same form as those obtained from [4] by substituting the accreditation interval durations into the two-class waiting time distribution given in Corollary 6.2.2 (in [4]), we take a different approach. (Obviously, the answer from either approach will be mathematically equivalent.) We consider the class \( k \) accredited busy period as a combination of class \( k - 1 \) delay cycles. Each is initiated by a customer served at priority level \( k \); such customers arrive at rate \( \sum_{i=1}^{k} \frac{b_k - b_{k+1}}{b_i} \lambda_i \) by Corollary 6.2.2. We denote the expected length of such a cycle by \( 1/\nu \).

\[ E(S) = \frac{1}{\mu_0(1 - \frac{1}{\nu} \sum_{i=1}^{k} \frac{b_i - b_{k+1}}{b_i})}. \] (6.50)

The first interval \( S_0 \) is just the first class \( k - 1 \) accreditation interval, which has length distribution \( D_{B_0}^{(k-1)} \) by Lemma 6.2.5. Substituting each of these parameters into Equation (6.49) above and scaling \( s \) by \( b_k \) to replace waiting time with priority shows that
\[ E(e^sV) = \frac{(\mu_0(1 - \frac{1}{\rho}\sum_{i=1}^k \lambda_i \frac{b_i - b_{k+1}}{b_i}))((\bar{D}^{(k)}_{B_0}(b_{k+1}s) - \bar{D}^{(k-1)}_{B_0}(b_k s))}{(1 - \frac{b_k}{b_{k+1}})(b_k s + \sum_{i=1}^k \lambda_i \frac{b_i}{b_i}(-1 + \bar{D}^{(k-1)}_{\beta_k}(b_k s)))}. \] (6.51)

### 6.4 Waiting time distribution

We now use the priority distribution from Theorem 6.3.1 to determine the waiting time distribution for an arbitrary class \( k \) customer, which we denote by \( W^{(k)} \).

A class \( k \) customer must either have arrived to an idle system, have queued and achieved accreditation relative to class \( k+1 \), or have queued and been serviced without becoming accredited relative to class \( k+1 \). A customer who arrives to an idle system experiences zero waiting time; this occurs with probability \( 1 - \rho \) by Theorem 2.5.2. For the other two cases, let \( V_a^{(k)} \) and \( V_u^{(k)} \) be the priority distribution for class \( k \) customers who enter during a busy period and are served while accredited or unaccredited relative to class \( k+1 \), respectively.

By Corollary 6.2.2, the proportion of class \( k \) customers who become accredited with respect to class \( k+1 \) is \( (b_k - b_{k+1})/b_k \); hence the remaining proportion \( b_{k+1}/b_k \) must be unaccredited.

\[
\tilde{W}^{(k)}(s) = 1 - \rho + \rho \left( \frac{b_k - b_{k+1}}{b_k} \right) \tilde{V}_a^{(k)}(s/b_k) + \rho \frac{b_{k+1}}{b_k} \tilde{V}_u^{(k)}(s/b_k)
\]
(6.52)
\[
= 1 - \rho + \rho \tilde{V}_+^{(k)}(s/b_k). \quad (6.53)
\]

Here \( \tilde{V}_+^{(k)} \) is the priority distribution function for those customers of class \( k \) who commence service with positive priority, or equivalently those who arrive during a busy period.

**Lemma 6.4.1.** Class \( k \) customers who are served while unaccredited relative to class \( k+1 \) have priority distributed as

\[
\tilde{V}_u^{(k)}(s) = \tilde{V}_+^{(k+1)}(s). \quad (6.54)
\]

**Proof.** By Theorem 6.1.2, the priorities of queued class \( k \) customers forms a Poisson process with rate \( \lambda_k/b_k \) on the interval \((0, M_k(t))\). A class \( k \) customer which is not accredited relative to class \( k+1 \) has priority less than \( M_{k+1}(t) \). Thus the priorities of queued class \( k \) customers who are unaccredited relative to class \( k+1 \) form a Poisson process, again with rate \( \lambda_k/b_k \), on the interval \((0, M_{k+1}(t))\), and both these customers and queued class \( k+1 \) customers in general have priorities uniformly distributed over \((0, M_{k+1}(t))\).

85
A customer is only chosen for service if they have more accumulated priority than all other queued customers, regardless of class. Thus, the priority of an unaccredited class \( k \) customer or arbitrary class \( k + 1 \) customer entering service will also have the same distribution, and the result follows.

A busy period may be decomposed into multiple class \( k \) accreditation intervals. (Recall that a class \( k \) accreditation interval is an interval in which each customer served is accredited with respect to class \( k + 1 \); this is the maximum accreditation level a class \( k \) customer can achieve.) Whenever a customer is served without having become accredited with respect to class \( k + 1 \), we say that they have begun a new class \( k + 1 \) accreditation interval.

Note that the case where \( k = 1 \) is not excluded here. No customer can be accredited with respect to class 1, so every customer to be served begins a new class 1 accreditation interval, and new accredited customers arrive at rate zero. Thus the duration of the accreditation interval is simply the service time of the first customer, which is distributed as an arbitrary class 1 customer service.

For a customer who is served at priority level \( k \) or higher, Theorem 6.3.1 gives us the distribution of priority over the level at which the class \( k \) accreditation interval started, given the initial delay of the class \( k \) accreditation interval. This initial delay will always be the service of a single customer, but the distribution of this customer’s class can differ. If all customer classes had the same priority distribution \( B \), the priority above the starting point of the accreditation interval would always be distributed as \( U_k \). However, we have not made that assumption, and as we will see, high priority class customers are more likely to start accreditation intervals which begin a busy period than accreditation intervals which occur later in a busy period, meaning that the initial delays of such periods have different distributions.

**Lemma 6.4.2.** Customers of class \( k \) who arrive to a busy system and are served at priority level \( k \) during a class \( k \) accreditation interval which began the busy period have priority distributed as \( \tilde{U}_k(s) \).

The proportion of class \( k \) arrivals who meet these criteria is

\[
\pi_0^{(k)} = \frac{\rho(1 - \rho)}{1 - \sigma_k}. \tag{6.55}
\]

**Proof.** If the system is idle, then any arrival will begin a class \( k \) accreditation interval. Such an arrival will be of class \( k \) with probability \( \lambda_k / \lambda \), and the service time of this customer will form the initial delay for the first class \( k - 1 \) accreditation interval. The service time distribution has LST

\[
\widetilde{B}(s) = \sum_{p=1}^{P} \frac{\lambda_p}{\lambda} B_p(s). \tag{6.56}
\]
This class $k$ accreditation interval will start from zero priority. Thus by Theorem 6.3.1, the priority distribution of a class $k$ customer served in such an interval is $\tilde{U}_k^B(s)$.

We must also determine the fraction of class $k$ jobs served at priority level $k$ who arrive to a busy system which is in its first class $k$ accreditation interval since the beginning of its busy period. By Theorem 4.1.2, the expected length of a delay cycle is given by $1/(\mu_I(1 - \rho_B))$, where $\mu_I$ is the expected length of the initial delay, and $\rho_B$ is the utilisation of blocking tasks.

Now the utilisation of blocking tasks is equivalent to the proportion of time the system spends on blocking tasks. Recalling that a blocking task in a class $k$ cycle is a job accredited with respect to class $k+1$, Theorem 6.2.6 tells us that the system spends a fraction $\sigma_k$ of its time on such tasks.

We also observe that the initial delay is a single service time, with the same class distribution as an arbitrary arrival, and hence it has expected duration $1/\mu$. Substituting into Theorem 4.1.2, we find that the expected length of a class $k$ accreditation interval starting from an idle system is given by $1/(\mu(1 - \sigma_k))$.

Finally, note that the system spends a proportion $1 - \rho$ of the time idle. Arrivals occur at rate $\lambda$, independently of the system state; hence arrivals to an idle system occur at rate $\lambda(1 - \rho)$. Any such arrival initiates a class $k$ accreditation interval. So the proportion of the time which the system spends on class $k$ accreditation intervals which begin from an idle system is

$$\pi_0^{(k)} = \frac{\rho(1 - \rho)}{1 - \sigma_k}. \quad (6.57)$$

Lemma 6.4.3. Customers of class $k$ who arrive to a busy system and are served at priority level $k$ during a class $k$ accreditation interval which did not begin the busy period and began with a class $j$ customer have priority distributed as

$$\tilde{V}^{(k+1)}(s)\tilde{U}_{B_j}^{(k)}(s), \quad j \leq k \quad (6.58)$$

$$\tilde{V}^{(j)}(s)\tilde{U}_{B_j}^{(k)}(s), \quad j > k. \quad (6.59)$$

The proportion of class $k$ arrivals who meet these criteria is

$$\pi_j^{(k)} = \begin{cases} \frac{\rho \rho_j}{1 - \sigma_j}, & j > k, \\ \frac{\rho \rho_j b_{k+1}}{b_j(1 - \sigma_j)}, & j \leq k \end{cases} \quad (6.60)$$

Proof. If the class $k$ accreditation interval does not begin a busy period, then it must be started by some customer who is not accredited relative to class $k+1$. If we assume that this customer has class $j$, then the distribution of priority beyond the level at which the class $k$ accreditation interval starts is given by Theorem 6.3.1 as $\tilde{U}_{B_j}^{(k)}(s)$. 

87
The initial delay of such a cycle simply has distribution function $B_j$, and mean $1/\mu_j$. The blocking tasks of such a cycle are again customers accredited with respect to class $k + 1$, and so again the utilisation of blocking tasks is given by $\sigma_k$. Thus, the expected length of the cycle is $1/(\mu_k(1 - \sigma_j))$.

Once the accreditation interval begins at some level $V$ with a customer of some class $j$, the extra priority which customers accumulate beyond that level is independent of $V$. (The higher the priority at which the accreditation interval begins, the more time that customers have had to queue and accumulate priority.) Recall that the Laplace-Stieltjes transform of the sum of two independent random variables is the product of the transforms of each random variable on its own.

No customer of class $j > k$ can be accredited relative to class $k + 1$, so the initiating customer will enter service with priority distributed as $V$. Thus, if $j > k$ the overall distribution of waiting time is given by $\hat{V}(s)\hat{U}_j^{(k)}(s)$.

Arrivals of class $j < k$ occur at rate $\lambda_j\rho$. (Recall that arrivals to an empty system have been considered earlier.) Multiplying by the mean cycle length, we see that the proportion of time which the system spends in such cycles is

$$\pi_j^{(k)} = \frac{\rho \rho_j}{1 - \sigma_j}, \quad j > k. \quad (6.61)$$

The class $k$ accreditation interval can also be initiated by a customer of class $j \leq k$, as long as this initiating customer is not accredited with respect to class $k + 1$. Such a customer commences service with priority identically distributed to class $k + 1$, by Theorem 6.1.2, so the priority at which the accreditation interval commences is distributed as $V^{(k+1)}$.

Hence in this case the overall distribution of waiting time is given by $\hat{V}^{(k+1)}(s)\hat{U}_j^{(k)}(s)$.

Customers of class $j \leq k$ arrive at rate $\lambda_j$ overall, and attain accreditation relative to class $k + 1$ during a busy period at rate $\lambda_j\frac{b_{k+1} - b_j}{b_j}$, by Lemma 6.2.1. Thus, the proportion of class $j$ customers who arrive during a busy period and are served without becoming accredited relative to class $k + 1$ is $b_{k+1}/b_j$. Accounting for idle periods as well, the proportion of all class $j$ arrivals who are served within a busy period while unaccredited relative to class $k + 1$ is $\rho b_{k+1}/b_j$. Each such customer initiates a class $k$ accreditation interval with expected length $1/(\mu_j(1 - \sigma_k))$. Thus, the proportion of time spent in such cycles is

$$\pi_j^{(k)} = \frac{\rho \rho_j b_{k+1}}{b_j(1 - \sigma_j)}, \quad j \leq k. \quad (6.62)$$

Stanford, Taylor and Ziedins observe that the sum of all the utilisation factors above is $\rho$, as expected [4].

□

88
Lemma 6.4.4. Class $k$ customers who are served during a busy period while accredited relative to class $k+1$ have priority distributed as

$$\tilde{V}_a^{(k)}(s) = \frac{\pi^{(k)}_0}{\rho} \tilde{U}^{(k)}_B(s/b_k) + \sum_{j=1}^{k} \frac{\pi^{(k)}_j}{\rho} \tilde{V}^{(k+1)}_B(s/b_k) + \sum_{j=k+1}^{P} \frac{\pi^{(k)}_j}{\rho} \tilde{V}^{(j)}(s/b_k) \tilde{U}^{(k)}_{B_j}(s/b_k).$$

(6.63)

Proof. Lemma 6.4.2 gives us the priority distribution for a class $k$ customer served during a class $k$ accreditation interval which begins a busy period; Lemma 6.4.3 covers other class $k$ accreditation intervals. Each lemma also gives us the proportion of time spent in each state, or equivalently, the probability that an arrival to a busy system will find the system in each state.

We simply take the sum of the priority distributions from Lemmas 6.4.2 and 6.4.3, weighted by the probabilities given in the same lemmas and then normalised by $\rho$, the proportion of the time in which the system is busy. This gives us the result above.

\[\Box\]

Theorem 6.4.5. The waiting time of a class $k$ customer has LST

$$\tilde{W}^{(k)}(s) = 1 - \rho + \frac{b_{k+1}}{b_k} \tilde{V}^{(k+1)}_+ (s/b_k) + \frac{b_k - b_{k+1}}{b_k} \tilde{V}^{(k)} (s/b_k) +$$

$$\frac{b_k - b_{k+1}}{b_k} \left( \sum_{j=1}^{k} \pi^{(k)}_j \tilde{V}^{(k+1)}_B(s/b_k) \tilde{U}^{(k)}_{B_j}(s/b_k) + \sum_{j=k+1}^{P} \pi^{(k)}_j \tilde{V}^{(j)}(s/b_k) \tilde{U}^{(k)}_{B_j}(s/b_k) \right).$$

(6.64)

Proof. The waiting time distribution for a class $k$ customer is given in Equation 6.52, in terms of the priority distributions for unaccredited and accredited customers. The former distribution is given in Lemma 6.4.1, and the latter in Lemma 6.4.4. Substituting these results into Equation 6.52 gives us the unconditional waiting time above.

\[\Box\]
Chapter 7

Numerical work

In this chapter we consider an accumulating priority queue with two classes, and calculate its waiting time distribution using the methods of the previous chapter.

Each class has arrival rate 1. Class 1 accumulates priority three times faster than class 2. The two classes share a service time distribution, which is exponential with rate 3. Observe that the queue is stable, with $\rho = 2/3$. The common service time distribution greatly simplifies our calculations, as all the class-dependent service distributions collapse to simply

$$\bar{B}(s) = \frac{\mu}{\mu + s}. \quad (7.1)$$

Moreover, the use of exponential service times means that we can use (or perhaps test) Kleinrock’s results for the mean waiting time from [7] and [23], as discussed in Chapter 5.

We implement the result of Theorem 6.4.5 in MATLAB to determine the Laplace-Stieltjes transform of the waiting time distribution. However, we modify it slightly to calculate $\tilde{W}_+$, the waiting time for a customer who arrives to a busy system. The reason for this is that our inversion method is intended for normal Laplace transforms rather than Laplace-Stieltjes transforms, so we want the waiting time to be differentiable. This does not pose a problem, as we are simply removing the point mass of $1 - \rho$ at the origin and it can be readily added back when needed. For example, when determining moments, we simply multiply the moment of $W_+$ by $\rho$.

Theorem 6.4.5 ultimately leads us back to the Laplace-Stieltjes transform of busy period durations given by Lemma 6.2.4. In general this expression is not analytically solvable. In this case, it is in fact possible to invert the expression, but in order to demonstrate the generality of our technique, we use a numerical solution. As per Wolff’s method from [13], we iterate the expression

$$\tilde{G}_{n+1}(s) = \bar{B}(s + \lambda - \lambda \tilde{G}_n(s)). \quad (7.2)$$
We use 20 iterations, and start from $\tilde{G}_1(s) = 0$; experimentation suggests that the initial condition is irrelevant, and 20 iterations gives adequate results.

Given the transform of the waiting time distribution, we want the corresponding probability density. We use the Gaver-Stehfest method [9, 10, 11], as described in Section 2.1, directly translating Equations (2.10) and (2.11) into MATLAB. This ultimately gives us a function which evaluates the density at any given point.

We calculate this density at intervals of 0.01, from 0.01 to 10. Note that the Laplace transform inversion in Equation (2.10) is undefined at zero. The results are shown in Figure 7.1. From here, we can numerically evaluate whatever properties of the distribution are of interest to us. We calculate the first two moments of waiting time, multiplying the results by $\rho$ to account for the point mass of $1 - \rho$ at the origin.

We also calculate the distribution function of waiting time for all arrivals, as shown in Figure 7.2. This function starts at $1 - \rho = 1/3$ at $t = 0$, and for positive $t$ it has derivative equal to the density function given above, scaled by $\rho$.

We display the mean and variance of waiting time for each class. We also use calculate the mean waiting times for this queue using Kleinrock’s formulae from [23, 7], from Equation (5.39). Sample output from the program follows:

Approximate integral of $W_1$ density: 0.99311
Approximate integral of $W_2$ density: 0.99363

$E(W_1)$ from ILT: 0.46943
$E(W_2)$ from ILT: 0.84647

$Var(W_1)$ from ILT: 0.3645
$Var(W_2)$ from ILT: 1.579

$E(W_1)$ by Kleinrock: 0.47619
$E(W_2)$ by Kleinrock: 0.85714

Firstly, note that the densities integrate very nearly to 1, which we find encouraging. Secondly, observe that the mean waiting times are very close to those calculated by Kleinrock. Given that there are two approximation methods involved in our calculations, it seems likely that Kleinrock's method is more accurate. However, while Kleinrock’s method can only give us the mean waiting time, we have the entire distribution. We show the variance as well; note that it is considerably higher for the low priority class.

The mode away from zero in class 1 (but not class 2) is interesting. We have no explanation for it at present.
Figure 7.1: Density of waiting time for arrivals to a busy accumulating priority queue
Figure 7.2: Distribution function of waiting times to an accumulating priority queue
Chapter 8

Conclusions

We have studied the M/G/1 queue with a number of priority disciplines. The method of busy periods, or delay cycles, was initially applied to the first-come first-served queue. However, extensions to this technique allow it to find the waiting time in the absolute priority and accumulating priority queues as well. The latter case also introduces the maximum priority process, giving us a much more tractable description of the behaviour of the accumulating priority queue than would otherwise be possible.

The method of delay cycles ultimately delivers results in the form of the Laplace-Stieltjes transform of a distribution function. Moreover, the transform is itself the solution of a functional equation, which usually does not have an analytic solution. We have demonstrated how to recover the waiting time distribution function from such a solution, showing the distribution of waiting time for an example accumulating priority queue. We also considered Kleinrock’s method for mean waiting times. This analysis is considerably simpler than the busy period approach, and thus may prove useful when considering more complicated queueing disciplines.

We have attempted to improve the rigour of our source material in several places. When considering the M/G/1 FCFS queue, we pursued the background results in renewal theory considerably further than Kleinrock in [6]. When considering the absolute priority queue, we attempted to remove the ambiguity in Conway, Maxwell and Miller’s definitions from [2]. We used Wald’s identity to improve the derivation of Kleinrock’s mean waiting time formulae. Finally, when considering the accumulating priority queue, we proved the main result for accumulated priority within accreditation intervals for \( k \) classes, rather than 2.

Overall, we have considered several queueing disciplines for the M/G/1 queue, shown how to obtain results for their waiting times, and shown how to convert the analytical transform results to more readily interpretable numerical forms.
8.1 Further work

There are a number of possible directions for further work in this area; we suggest a few in this section.

Many priority disciplines have been suggested, of which we have only considered three. There are surely more which are amenable to the kind of analysis performed in the previous chapters.

Another obvious extension is to the case of multiple servers. However, the stationary distribution of the M/G/k queue has not been solved for any priority discipline [1]. It seems that any solution will require new techniques, and a solution for more complicated priority disciplines may be even further away. Similarly, our extensive use of Poisson arrivals means that the techniques considered here are unlikely to generalise to the GI/G/1 case.

Another possible extension is to preemptive queues. It seems that the methods discussed here could be more readily applied here. In the absolute priority case, the highest priority class would simply ignore all lower classes. Other classes could be interrupted, and any interruption would cause a busy period of higher priority customers, after which the lower priority class would re-enter service. There would be some additional complications for preempt repeat queues; in such cases a customer would only complete service if no higher priority customer arrived during their entire service interval. This does not appear beyond the reach of the techniques already discussed here, however. A preemptive accumulating priority queue may need some further restrictions to prevent customers from displacing each other from service infinitely often, but again the busy period analysis seems applicable.

We believe that the maximum priority process is worthy of further consideration. This process, or some similar construction, may be useful in other queueing systems. We would also be interested in its stationary distribution, both for its theoretical value and because it may provide further insight into the accumulating priority queue.

A final area of extension is the numerical methods used to obtain. We listed three methods for inverting Laplace transforms, but only implemented one. It would be worth examining which is most suitable for the transforms found here. Similarly, more sophisticated methods for approximating the solution to the busy period functional equation would result in more accurate distributions.
Bibliography


