Stationary Waiting Time Distribution of the Maximum Priority Process in the Accumulating Priority M/M/1 Queue

D.J.J. Dams
Stationary Waiting Time Distribution of the Maximum Priority Process in the Accumulating Priority $M/M/1$ Queue

D.J.J. Dams - 0636842

MN-420751

Internship report

Supervisors: prof. dr. P.G. Taylor (The University of Melbourne)
prof. dr. ir. I.B.J.F. Adan

THE UNIVERSITY OF MELBOURNE
DEPARTMENT OF MATHEMATICS AND STATISTICS

EINDHOVEN UNIVERSITY OF TECHNOLOGY
DEPARTMENT OF MECHANICAL ENGINEERING

Eindhoven, January 13, 2014
Preface

I obtained my Bachelor degree in Mechanical Engineering at Eindhoven University of Technology in 2011. After that, I joined the Manufacturing Networks group from the Dynamical Systems Design Master track at the same university to start my Master.

A part of my Master is an internship. I made the choice to go abroad and to participate in a research project in the Applied Probability group from the Department of Mathematics and Statistics at the University of Melbourne in Australia, led by professor Peter Taylor. This report concludes my internship and research I carried out on the stationary waiting time distribution of the maximum priority process in the accumulating priority $M/M/1$ queue. I started my internship in Melbourne in November 2012 and finished the research at the end of March 2013.

First of all, I would like to thank professor Peter Taylor, who has supervised me during my project in Melbourne. His remarks and comments on my work were a welcome addition to carry out the research better and to make me understand the whole principle of the maximum priority process.

Next, I would like to thank professor Ivo Adan for the support he gave me from Eindhoven and dr.ir. Marko Boon who gave me an introduction into Mathematica. Which was very important for our attempts to solve the complex expressions symbolically.

Finally, I would like to thank my family for supporting my stay in Australia and Michael Metz, Nick Metz, William Grbic, Georgie Smibert, Jay Wu, Sam Huang and Art Dejmanee for showing me an amazing time.
Summary

In this report we extend Kleinrock’s [2], David A. Stanford’s, Peter Taylor’s and Ilze Ziedins’ [1] analysis on the accumulating priority queue. Therefore, we first introduce the accumulating priority queue. In classical queueing theory the virtual workload process is used for deriving the stationary waiting time process. In our analysis we make use of the maximum priority process, we make the connection between the virtual workload and maximum priority process. Furthermore, observations in the two class accumulating priority queue are made.

After the introduction of the accumulating priority queue, a study of the one class maximum priority process in the accumulating priority $M/M/1$ queue is provided. In this study we derive a framework, which makes use of the Laplace transform, to obtain the stationary waiting time distribution of the queue with analysis of the maximum priority process. Our framework is also a new way to derive the stationary waiting time distribution in the $M/M/1$ queue.

The derived framework is applied on the two class accumulating priority $M/M/1$ queue. Following the steps we defined in our study on the one class queue we try to obtain the stationary waiting time distribution. We are able to derive the expressions for the limiting distributions in the Laplace domain for the two class queue. These expressions are basically the stationary waiting time distributions with some unknown functions that we try to solve by substitution. For the unaccredited customers a 1D distribution should be obtained while the waiting times of the accredited customers are described by a 2D distribution. However, until now, we are not able to actually solve the expressions and obtain the stationary waiting time distributions we want to obtain.

Analytical derivation of the stationary waiting time distributions in the two class accumulating priority $M/M/1$ queue is not provided in this report. We are able to obtain the stationary waiting time distribution numerically. Therefore, a simulation procedure is developed which makes also use of the maximum priority process. With this simulation procedure also some m-files are provided for the numerical inversion of complex distributions in the Laplace domain. These m-files could be of assistance in further research on the accumulating priority queue. As a conclusion to the report we provide the reader with our conclusions and some recommendations for further research.
# Contents

1 Introduction .......................................................... 1

2 The accumulating priority queue .................................. 3
   2.1 Accumulating priority model ................................ 3
   2.2 Maximum waiting time process in the single class FCFS queue .... 5
   2.3 Maximum priority process in the two class accumulating priority queue .... 7
   2.4 Observations on the maximum priority process in the two class accumulating priority queue ............................... 8

3 Study of the one class maximum priority process in the $M/M/1$ queue 11
   3.1 Markov Chain of the one class maximum priority process ............ 11
   3.2 Derivation of the limiting distributions of the one class problem ........ 15

4 Study of the two class maximum priority process in the $M/M/1$ queue 21
   4.1 Markov Chain of the two class maximum priority process ............ 21
   4.2 Expressions for the limiting distributions of the two class maximum priority process 30
   4.3 Transformation of the expressions for the limiting distributions to the Laplace domain ........................................ 32
   4.4 Limiting distributions in the Laplace domain ....................... 35

5 Numerical tools .......................................................... 39
   5.1 Simulation procedure ............................................ 39
   5.2 Numerical inversion of Laplace transforms .......................... 44

6 Conclusion and recommendations ...................................... 49
   6.1 Conclusion ......................................................... 49
   6.2 Recommendations .................................................. 50
Bibliography

A  Proof of the convolution theorem  57

B  Check for transition probability and density functions in the two class problem  59

C  Working out the integrals in right hand side of $\pi(x, x)$  63

D  Working out the integrals in right hand side of $\pi(x, y)$  67

E  Matlab code: simulation file  71

F  Matlab code: determining waiting times customers  79

G  Matlab code: probability density file  81

H  Matlab code: Euler method algorithm  83

I  Matlab code: Post-Widder method algorithm  87
Chapter 1

Introduction

The traditional way to analyze priority queues is to assume that the different customer classes have fixed priorities. Besides that, a customer is not allowed service while customers of a higher class are present in the system. In many situations, this type of priority queueing discipline is appropriate. However, it is possible to think of a situation where this discipline is not appropriate, consider for instance a situation where separate service requirements are specified for each customer class. An absolute priority discipline will not yield performance levels that satisfy the service requirements. For example, high priority classes might receive better service than specified while the service level of lower classes might not be adequate. Therefore, it is desirable to seek a modification to the traditional structure of fixed priorities, with this modification the manager should be able to fine-tune the customer selection discipline so that the service requirements of all customer classes are simultaneously satisfied.

In 1964, Kleinrock [2] proposed a discipline called the time-dependent priority queue, a simple discipline for achieving such an objective. Kleinrock’s goal was to achieve desired ratios of stationary mean waiting times experienced by customers from the different classes. He achieved this by stipulating that customers accumulate priority as a linear function of their time in the queue, with customers from classes for whom mean waiting time should be shorter and thus accumulating priority at a greater rate. When service of a customer is finished and he leaves the server, the next customer to be served (if any in the queue) is the one with the highest accumulated priority at that time point.

Recently, David A. Stanford, Peter Taylor and Ilze Ziedins [1] recognized that the performance of many queues, particularly the ones in health care and human services sectors, is specified in terms of tails of waiting time distributions for customers of different classes. In health care applications, patients in many jurisdictions around the world are classified according to an equity rating system. The performance of such systems is assessed in terms of Key Performance Indicators (KPIs) expressed in terms of distributional tails. These KPIs specify, for the different classes, both a benchmark time standard and a proportion of patients whose waiting times should not exceed the stipulated standard. An example is provided in Table 1.1, in this table the Canadian Triage and Acuity Scale is depicted.
Table 1.1: Canadian Triage and Acuity Scale Key Performance Indicators [1]

<table>
<thead>
<tr>
<th>Category</th>
<th>Classification</th>
<th>Access</th>
<th>Performance level</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Resuscitation</td>
<td>Immediate</td>
<td>98 %</td>
</tr>
<tr>
<td>2</td>
<td>Emergency</td>
<td>15 minutes</td>
<td>95 %</td>
</tr>
<tr>
<td>3</td>
<td>Urgent</td>
<td>30 minutes</td>
<td>90 %</td>
</tr>
<tr>
<td>4</td>
<td>Less urgent</td>
<td>60 minutes</td>
<td>85 %</td>
</tr>
<tr>
<td>5</td>
<td>Not urgent</td>
<td>120 minutes</td>
<td>80 %</td>
</tr>
</tbody>
</table>

Stanford et al derived the waiting time distributions of the different customer classes in a queueing discipline as explained in the previous paragraph, with Poisson arrivals and generally distributed service time distributions, thus an $M/G/1$ queueing system [1]. They refer to the discipline as the *accumulating priority* queue, since the term *time-dependent priority* queue has come to mean many things. In the study of Stanford et al the introduction of a new stochastic process called the *maximum priority* process is made.

We are interested in the stationary waiting time distribution of this *maximum priority* process in the accumulating priority queue. Eventually, the stationary waiting time distribution should be obtained for an $M/G/1$ queueing system in which the different customer classes receive different generally distributed service times. However, until now it is not known how to derive this stationary waiting time distribution. Therefore, we start with a study of the maximal priority process in an $M/M/1$ queue with one customer class. Since the stationary waiting time of such system is known we can verify the method we develop. The maximum priority process is a continuous-time process, by only taking the embedded time points where customers enter service into consideration we are able to set up a Markov Chain. With this Markov Chain we can derive expressions for the limiting distributions of the system. To solve these expressions we introduce a method which makes use of the Laplace transform. Our next step is to apply our method on the maximum priority process in a two class accumulating priority $M/M/1$ queueing system.

In this report we extend Kleinrock’s, David A. Stanford’s, Peter Taylor’s and Ilze Ziedins’ analysis. We derive the stationary waiting time distribution of the maximum priority process in the one class accumulating priority $M/M/1$ queue. With the approach we use in the one class queue we try to solve the stationary waiting time distributions of the maximum priority process in the two class accumulating priority $M/M/1$ queue.

In Chapter 2 the accumulating priority queue model we analyse is explained in more detail. After this chapter we discuss the study of the maximum priority process in the one class accumulating priority $M/M/1$ queue in Chapter 3. The next step is to apply this newly developed and discussed method to derive the stationary waiting time distributions of the maximum priority process in the two class accumulating priority $M/M/1$ queue, the reader will find this study in Chapter 4. In Chapter 5 we discuss two numerical tools we use to analyze the accumulating priority queue, the first tool is a simulation program that simulates the maximum priority process in the two class $M/M/1$ queue and the second tool a numerical inversion of Laplace transforms file. We conclude this report with Chapter 6, here we will provide comments on our method and suggestions for further research are made.
Chapter 2

The accumulating priority queue

Stanford, Taylor and Ziedins [1] introduced a new stochastic process called the maximum priority process. This stochastic process was proven to be very useful in the study of the accumulating priority queue.

In this chapter we shall first discuss the model of the accumulating priority queue that is considered by Stanford et al. When this model is explained a new way of deriving the waiting time distribution of a queueing model, with the maximum waiting time process, is introduced. From the maximum waiting time process we arrive at the stochastic process that we call the maximum priority process. This process is defined and explained with an example of a two class accumulating priority queue. After that we point out some important observations about the two class accumulating priority queue.

2.1 Accumulating priority model

Stanford et al considered a single server $M/G/1$ queue with Poisson arrivals and generally distributed service times. Customers arriving at this queue have different classes, $i = 1, 2, ..., N$, with the arrival rate of class $i$ denoted by $\lambda_i$. When a customer enters the queue it starts accumulating priority with a certain priority rate $b_i$. The accumulating priority rate of a higher class customer will be greater than the rate at which a lower class customer accumulates priority, thus $b_1 > b_2 > ... > b_N$. We illustrate the priority accumulation with an example. Consider a customer of class $i$, this customer enters the queueing system at time point $t_1$ and service is allowed at time point $t_2$. The accumulated priority of this customer when it enters service is then equal to $b_i(t_2 - t_1)$.

When the server becomes idle and if there are customers waiting in the queue, the customer with the greatest accumulated priority is taken into service. This customer is not necessarily a customer of the highest customer class (which occurs when we take a priority queue with fixed priorities into consideration). This is because there could be already a lower class customer present in the queue for a significantly longer time than a higher class customer at the moment that customer enters the queue. So, at the specific time point the lower class customer has a
higher accumulated priority than the higher class customer.

A plot of the accumulating priority process is depicted in Figure 2.1. In the figure a sample path of the process with two customer classes \((i = 1, 2)\) is depicted. The class 1 customers accumulates priority with rate \(b_1 = 1\) (the blue colored path) and a class 2 customer accumulates priority with rate \(b_2 = 0.5\) (the green colored path). It can be seen that the sample path is plotted for a system which handled five customers. There are three arrivals of class 1 customers at time points \((1, 10, 17)\) and at time points \((3, 15)\) we have arrivals of class 2 customers. When the first customer enters the system he encounters an empty queue and an idle server, so at time point 1 this customer is directly taken into service. During the service of the first customer we can see an arrival of a second, who is of class 2, and a third, who is of class 1, customer. At the time point the server finishes service of the first customer (time point 14) we see that the class 2 customer has a higher priority then the class 1 customer. So, the second customer that is allowed service is a class 2 customer, who was the second customer to arrive at the queue.

During service of the second customer we observe arrivals at time points 15 and 17 of a class 2 and class 1 customer, respectively. Another interesting aspect that can be observed during the service of the customer which arrived second at the queue, is the overtaking of a class 2 customer by a class 1 customer at time point 19. So, after the server finishes the second customer it serves the next customers in the order of the third, fifth and fourth customer who arrived at the queue.

To make a proper analysis of the accumulating priority model we describe in this section we need to define a couple of stochastic primitives. We introduce \(X_n, T_n\) and \(\chi(n)\) which denotes the service times of the customer, the inter-arrival times of the customers entering the queue and the specific class of the customer, with stochastic processes \(X = \{X_n; n = 1, 2, ...\}\), \(T = \{T_n; n = 1, 2, ...\}\) and \(\chi = \{\chi(n); n = 1, 2, ...\}\), respectively.
2.2 Maximum waiting time process in the single class FCFS queue

The traditional way of analyzing queueing systems is with the so-called virtual workload process [2]. With this process it is possible to derive the waiting times and busy period distributions of the system that is considered. A new way of analyzing queueing systems, called the maximum waiting time process, was introduced by Stanford et al [1]. This maximum waiting time process shall be connected in the next section to the accumulating priority queue.

However, first the connection between the virtual workload and maximum waiting time processes needs to be made. Therefore an M/G/1 FCFS queue with one customer class is taken into consideration. The customers have an arrival rate $\lambda$ and their service times have mean $1/\mu < \infty$ with $\lambda < \mu$, to ensure stability.

The virtual workload process $U = \{U(t); t \geq 0\}$ measures the amount of work remaining in the queue at any time $t$ [2]. In terms of the arrival and service processes, this process is defined as

$$U(t) = \left( \sum_{n=1}^{N_A(t)} X_n - t \right) - \min_{s \leq t} \left( \sum_{n=1}^{N_A(s)} X_n - s \right)$$

(2.1)

where $X_n$ denotes the service time of the $n^{th}$ customer and $N_A(t)$ denotes the number of arrivals that have occurred by time $t$ and is defined as

$$N_A(t) = \max\{j : \sum_{k=1}^{j} T_k \leq t\}. \quad (2.2)$$

It is also possible to define a process which is equal to zero at the time point $t$ if there are no customers present in the system. When this is not the case this process takes the maximum possible waiting time of any customer still present in the queue at time $t$, given the history of the process up to the time that the current customer started service taken into account [1]. The described process is called the maximum waiting time process and can be expressed as

$$W_{\text{max}}(t) = [t - \sum_{k=1}^{N_S(t-L(t))} T_k]^+ = [t - \tau_{N_S(t-L(t))}]^+$$

(2.3)

where $T_k$ denotes the inter-arrival time of the $k^{th}$ customer. So, $\tau_n = \sum_{k=1}^{n} T_k$ denotes the time of the $n^{th}$ arrival. Furthermore, $N_S(u)$ denotes the maximum number of customers who would have commenced service at time $u$ if the system had not experienced any idle time [1]. $L(t), t \geq 0$ denotes the cumulative idle time experienced by the server up to time $t$. $N_S(u)$ and $L(t)$ are expressed as

$$N_S(u) = \min\{j : \sum_{m=1}^{j} X_{n(m)} > u\}$$

(2.4)
Chapter 2. The accumulating priority queue

and

\[ L(t) = \int_0^t I\left[ \sum_{k=1}^{N_\ell(u-L(u))} T_k \geq u \right] du, \]

respectively [1].

The connection between the virtual workload process and the maximum waiting time process is illustrated in Figure 2.2 [1]. In this figure \( X_i, T_i \) and \( W_i \), with \( i = 1, 2, 3, 4, 5 \), denote the service, inter-arrival and waiting times of customer \( i \), respectively. First, we take the virtual workload process which we find in the upper plot in consideration.

At a certain time point the \( n^{th} \) customer will arrive at an empty system and initiate a busy period, we observe a jump of \( X_n \), so at this time point the work that needs to be carried out by the server in the system is equal to the service time of the \( n^{th} \) customer. From that point the workload of the system will gradually decrease until the queue becomes empty or the time point the \( (n+1)^{st} \) customer arrives at the system, which is equal to the time where the \( n^{th} \) customer commenced service plus the inter-arrival time \( (T_{n+1}) \) of the \( (n+1)^{st} \) customer. At this time point the service time \( X_{n+1} \) of the customer is added to the total workload in the system. Since, \( X_n \) and \( T_{n+1} \) are known we immediately know the waiting time of the \( (n+1)^{st} \) customer \( (W_{n+1}) \), which is depicted in the figure with a red line, because \( W_{n+1} = X_n - T_{n+1} \). As long as the busy period remains this is repeated. If the inter-arrival time of the \( (n+1)^{st} \) customer is greater than the waiting time plus the service time of the \( n^{th} \) customer the busy period finishes and an idle period starts.

Next we consider the maximum waiting time process. This process is depicted in the lower plot of Figure 2.2. In this plot we observe that the maximum possible waiting time at any time point is plotted. If a customer (say the \( n^{th} \)) arrives in an empty system the maximum possible waiting time is clearly equal to zero and the customer immediately commences service. As long as the \( n^{th} \) customer is in service there could arrive an \( (n+1)^{st} \) customer in the queue. Since it is possible for the \( (n+1)^{st} \) customer to arrive shortly after the arrival of the \( n^{th} \) customer the maximum possible waiting time gradually increases during the service time of the first customer which is \( X_n \). At the time point that the server finishes the service of the \( n^{th} \) customer and if a \( (n+1)^{st} \) customer arrived in the system during its service, the maximum possible waiting time will drop to a certain point. The measure of this drop is equal to the inter-arrival time of the \( (n+1)^{st} \) customer \( (T_n) \). The point to which the maximum possible waiting time drops is equal to the waiting time of the \( (n+1)^{st} \) customer which is again equal to \( W_{n+1} = X_n - T_{n+1} \). This process continues until the inter-arrival time of the next customer is greater than the maximum possible waiting time which is build up in the system. Now, the maximum waiting time shall drop to zero, it stays zero during the whole idle period of the system, this is obvious since there cannot be any waiting time in the system when there is no customer present in the queue. Now, an idle period will start. This period finishes when a new customer enters the system and initiates the next busy period.

So, we can analyze the waiting time process of the queueing system with one customer class either with the virtual workload or the maximum waiting time process. We prefer the latter for the analysis of the one class accumulating priority queue. The reason for this is that the
2.3 Maximum priority process in the two class accumulating priority queue

At this point we have enough information for the introduction of the maximum priority process in a two class accumulating priority queue. Stanford et al [1] introduced the accumulating priority function for the \( n \)th customer which is defined by

\[
V_n(t) = \begin{cases} 
 b_{\chi(n)}[t - \tau_n]^+ & \text{if } \tau_n < t \\
0 & \text{otherwise} 
\end{cases}
\]  

(2.6)

In this expression \( \chi(n) \) denotes the class of the \( n \)th customer, for the two class accumulating priority queue \( \chi(n) \) can either be 1 or 2. For the ease of notation, the priority of a customer
Chapter 2. The accumulating priority queue

keeps accumulating during their service and after departure.

In Section 2.1 we already saw that the accumulating priority queue is not a FIFO system. Therefore, Stanford et al [1] defined \( n(m) \) to be the position in the arrival sequence of the \( m^{th} \) customer to be served. So, for instance, if the 10\(^{th} \) arrival was actually the 4\(^{th} \) to be served, then \( n(4) = 10 \). Furthermore, let \( C_n \) be the time at which service commences for the \( n^{th} \) customer to arrive. So, for instance, if the 10\(^{th} \) arrival was actually the 4\(^{th} \) to be served, then \( n(4) = 10 \). Furthermore, let \( C_n \) be the time at which service commences for the \( n^{th} \) arrival. So, the time at which the \( m^{th} \) service commences is given by \( C_n + X_n \) is the departure time of the \( n^{th} \) customer to arrive, with \( C = \{ C_n; n = 1, 2, \ldots \} \) and \( D = \{ D_n; n = 1, 2, \ldots \} \). We can further write that \( C_n(1) = C_1 = T_1 = \tau_1 \) and \( C_n(m+1) = \max\{D_{n(m)}, \tau_{n(m+1)}\} \).

Now the maximum priority process for the accumulating priority queue in the two class case can be defined [1]:

**Definition 1.** The maximum priority process \( M = \{(M_1(t), M_2(t)), t \geq 0\} \) for the accumulating priority queue in the two class case is defined as follows.

1. At the sequence of departure times \( \{D_{n(m)}, m = 1, 2, \ldots \} \),

\[
M_1(D_{n(m)}) = \max_{n \in \{n(i) : 1 \leq i \leq m\}} V_n(D_{n(m)})
\]

\[
M_2(D_{n(m)}) = \min\{M_1(D_{n(m)}), M_2(C_n(m)) + b_2 X_{n(m)}\}.
\]

2. For \( t \in [C_n(m), D_{n(m)}) \) with \( \max_{n \in \{n(i) : 1 \leq i \leq m\}} V_n(t) > 0 \) (that is, when there are customers present in the queue),

\[
M_i(t) = M_i(C_n(m)) + b_i(t - C_n(m)).
\]

We note that in the above, \( C_n(m) = \max\{\tau_{n(m)}, D_{n(m-1)}\} \). If the queue is empty at time \( t \), then \( M_1(t) = M_2(t) = 0 \).

The underlying idea of this process is that, for each time \( t \geq 0 \) which is not a departure time and for each class of customer, it gives the least upper bound for the priorities of customers for each class, given only knowledge of the time at which the most recent service started, and the accumulated priority of the customer currently in service at the time point its service commenced. At the departure time points, \( M_1(t) \) is determined by the maximum of the accumulated priority of the customer which is the next to commence service. This is depicted in Figure 2.3, this figure plots the maximum priority process (in bold) against the time \( t \) for the sample path shown in Figure 2.1, superimposed on the priority functions \( V_n(t) \) [1].

### 2.4 Observations on the maximum priority process in the two class accumulating priority queue

In the paper from Stanford et al [1] numerous observations of the maximum priority process are made. The observations that are most important in our work to obtain the stationary waiting time distribution of the accumulating priority queue are pointed out here.
2.4. Observations on the maximum priority process in the two class accumulating priority queue

The first important observation Stanford et al made comes with the assumption that the arrival process is Poisson. This process leads to a result that can be exploited to show that the distributional properties of the maximum priority process are preserved if we do not keep track of the accumulated priority of the waiting customers. Instead, we sample the maximum of the accumulated priority at each departure point. To express this Stanford et al defined $M(t) \equiv \sigma\{ (M_1(u), M_2(u)) : u \in [0,t] \}$ to be the filtration generated by the maximum priority process up to time $t$. With this filtration Theorem 3.2 in Stanford et al states

\textbf{Theorem 1.} Let $t \in [0, \infty)$. Then, conditional on $M(t)$, the accumulated priorities $V_n(t)$ of the customers still present in the queue, but not yet in service, are distributed as a non-homogeneous Poisson process with piecewise constant rates zero on the interval $(M_1(t), \infty)$, $\lambda_1 b_1$ on the interval $(M_2(t), M_1(t))$ and $\frac{\lambda_1 b_1}{b_2} + \frac{\lambda_2 b_2}{b_1}$ on the interval $[0, M_2(t)]$. The statement holds also for any random time $T$ that is a stopping time.

For the proof of this theorem the reader is directed to page 9 of Stanford et al [1].

Another important observation is the difference between customers that have accumulated priorities at time $t$ that lie in the intervals $(M_2(t), M_1(t))$ and $[0, M_2(t)]$. The customers that have their accumulated priorities at time $t$ in the interval $(M_2(t), M_1(t))$ are called the \textit{accredited} customers. An accredited customer necessarily is of class 1 since its accumulated priority is greater than $M_2(t)$, which is the upper bound of the accumulated priority of a class 2 customer. Furthermore, since $b_1 > b_2$ we know that once a customer becomes accredited, they remain accredited until they enter service. Thus an accredited class 1 customer receives guaranteed service before any waiting class 2 customer.

The customers that have their accumulated priorities at time $t$ in the interval $[0, M_2(t)]$ are
called the *unaccredited* customers. An unaccredited customer can be either a class 1 or a class 2 customer. Consider a customer with priority \( v \in [0, M_2(t)] \) at time \( t \). If such an unaccredited customer is of class 2 its waiting time has been \( \frac{v}{b_2} \). If this customer is a class 1 customer its waiting time is equal to \( \frac{v}{b_1} \). The priorities of customers lying in the interval \([0, M_2(t)]\) are distributed according to a Poisson process with rate \( \frac{\lambda_1 b_1}{b_1} + \frac{\lambda_2 b_2}{b_2} \), see Theorem 1. These priorities are generated by a mixture of class 1 and 2 customers. The unaccredited class 1 customers arrived over the time interval \((t - M_1(t), t]\) and have accumulated priority at rate \( b_1 \) and the class 2 customers arrived over the time interval \((t - M_2(t), t]\) and accumulated priority with rate \( b_2 \). By Theorem 1, the distribution of the priorities at time \( t \) is the same as if customers had arrived in a Poisson process with rate \( \lambda_2 + \frac{\lambda_1 b_2}{b_1} \) over the whole interval \((t - \frac{M_2(t)}{b_2}, t]\) and had all been accumulating priority at rate \( b_2 \).

The last observation that we make is that the periods where \( M_1(t) = M_2(t) = 0 \) correspond to idle periods of the queue. The durations of these periods are independent and exponentially distributed with parameter \( \lambda_1 + \lambda_2 \). From classical queueing theory we know that the stationary probability that \( M_1(t) = M_2(t) = 0 \) is equal to \( 1 - \rho \).

For more observations of the maximum priority process, and then especially for the so-called accreditation intervals, the reader is directed to the paper of Stanford et al [1].
Chapter 3

Study of the one class maximum priority process in the $M/M/1$ queue

We consider the maximum priority process in an $M/M/1$ queueing system with one customer class. In the analysis of this system, $\lambda$ and $\mu$ denote the Poisson arrival and service rate, respectively. The utilization is denoted with $\rho = \frac{\lambda}{\mu}$ and the priority rate at which the customer accumulates priority is equal to $b$. Since there is only one customer class present we choose $b = 1$. Therefore, the reader will not find the parameter $b$ in the expressions in this chapter.

This chapter contains two sections. In the first section we define the transition probabilities and densities of a Markov Chain nested for the maximum priority process at the times at which customers enter service. With these probability and density functions the expressions for the limiting distributions of this Markov model are derived. In the last section we derive the stationary waiting time distribution of the one class maximum priority process by transforming the expressions for the limiting distributions to the Laplace domain.

3.1 Markov Chain of the one class maximum priority process

The maximum priority process is a continuous-time process. The discrete time process ($M^n$) that we study is derived from this process by looking at the embedded points where customers enter service. This process is depicted schematically in Figure 3.1 in a Markov Chain (MC). The MC of the system contains; (i) the state at which the server is idle and (ii) a 1-D continuum set of states at which a customer, having accumulated priority enters service. In the figure this set of states is depicted with the red dashed line around the Service Customer box. This box also indicates the set where the model is in the busy period.
Chapter 3. Study of the one class maximum priority process in the $M/M/1$ queue

This continuous-state Markov process has a transition probability into the Idle state (0) and a transition density into the Service Customer set ($x$), with priority $x \in (0, \infty)$, from all states present in the MC. In total we distinguish four different types of transitions in this continuous-state Markov process, i.e. $(0) \to (0)$, $(v) \to (0)$, $(0) \to (x)$ and $(v) \to (x)$, in which $v \in (0, \infty)$ and $x \in (0, \infty)$ denote accumulated priorities. The transition probabilities and densities belonging to the MC are

1. **Idle period to Idle period:** When this transition occurs we know that the single service of duration $u \in (0, \infty)$ received by a customer that arrives to an empty queue finishes before any arrival of a new customer occurs. So, the accumulated priority at the moment the service is finished is equal to 0. A schematic representation of this transition is depicted in Figure 3.2a. By conditioning on $u$, the transition probability is given by

$$P(M^{n+1} = 0 | M^n = 0) = \int_0^\infty \mu e^{-\mu u} \lambda e^{-\lambda u} du = \frac{\mu}{\lambda + \mu}. \quad (3.1)$$

In Expression (3.1) we recognize the “race” in the $M/M/1$ queue between the service and arrival processes, which in this case the service wins. For the ease of notation throughout the rest of this report we shall refer to this expression as $p_{0 \to 0}$.

2. **Service customer to Idle period:** When this transition occurs, a customer, which waited in the queue before it commenced service and thus accumulated priority, is served. In the interval before it started service and during its service no arrival of a new customer occurs. The priority $v \in (0, \infty)$ denotes the accumulated priority at the moment the customer commenced service. The duration of service is again denoted with $u \in (0, \infty)$. A schematic representation of this transition is depicted in Figure 3.2b. By conditioning on $u$, the transition probability function is given by

$$P(M^{n+1} = 0 | M^n = v) = \int_0^\infty \mu e^{-\mu u} \lambda e^{-\lambda (u+u)} du = e^{-\lambda v} \frac{\mu}{\lambda + \mu}. \quad (3.2)$$

Throughout the rest of this report this expression is denoted with $p_{v \to 0}$.

![Figure 3.1: The Markov Chain of the maximum priority process with one customer class.](image-url)
3.1. Markov Chain of the one class maximum priority process

3. Idle period to Service customer: When this transition occurs, the \((m+1)^{st}\) customer arrives during the service of the \(m^{th}\) customer who initiated a busy period. So, the priority \(x \in (0, \infty)\) denotes the accumulated priority at the moment the \((m+1)^{st}\) customer commences service. A schematic representation of this transition is depicted in Figure 3.2c. By conditioning on the length of the interval between the \(m^{th}\) and \((m+1)^{st}\) customer (thus again the service \(u \in (0, \infty)\)) we see that the transition probability density function is given by

\[
P(M^{n+1} = x | M^n = 0) = \int_0^\infty \lambda e^{-\lambda u} \mu e^{-\mu(u+x)} du = e^{-\mu x} \frac{\lambda \mu}{\lambda + \mu}. \quad (3.3)
\]

Throughout the rest of this report this expression is denoted with \(f_{0 \to x}\).

4. Service customer to Service customer: When this transition occurs, the \((m+1)^{st}\) customer of the busy period arrives during service \(u \in (0, \infty)\) of the \(m^{th}\) customer (with \(m = 2, 3, \ldots\)) in the busy period. In his turn, when the \(m^{th}\) customer arrived at the system he had to wait for the service of the \((m-1)^{st}\) customer to be finished, before his own service commenced. Schematic representations of these transitions are depicted in Figures 3.2d and 3.2e. As can be seen in these plots, a distinction is made between two possibilities, i.e. \(x \geq v\) and \(x < v\).

(a) when \(x \geq v\), by conditioning on the length of the inter-arrival time between the \(m^{th}\) and \((m+1)^{st}\) customers (thus service duration \(u\)), we see that for \(v \in (0, \infty)\) and \(x \in [v, \infty)\)

\[
P(M^{n+1} = x | M^n = v) = \int_0^\infty \lambda e^{-\lambda u} \mu e^{-\mu(u+v+x)} du = e^{-\mu(x-v)} \frac{\lambda \mu}{\lambda + \mu}. \quad (3.4)
\]

Throughout the rest of this report Expression (3.4) is denoted by \(f_{v \to x | x \geq v}\).

(b) when \(x < v\), again by conditioning on the length of the service time and for \(v \in (0, \infty)\) and \(x \in (0, v)\) the probability density function is given by

\[
P(M^{n+1} = x | M^n = v) = \int_0^\infty \lambda e^{-\lambda(v+u-x)} \mu e^{-\mu u} du = e^{\lambda v} \frac{\lambda \mu}{\lambda + \mu}. \quad (3.5)
\]

Throughout the rest of this report Expression (3.5) is denoted by \(f_{v \to x | x < v}\).

The total transition probability out of every state should be equal to one. Thus we expect that

\[
p_{0 \to 0} + \int_0^\infty f_{0 \to x} dx = 1, \quad (3.6)
\]

and

\[
p_{0 \to 0} + \int_0^v f_{v \to x | x < v} dx + \int_v^\infty f_{v \to x | x \geq v} dx = 1. \quad (3.7)
\]

The left hand side of Expression (3.6) is equal to
Chapter 3. Study of the one class maximum priority process in the $M/M/1$ queue

(a) An idle period to idle period transition

(b) A service customer to idle period transition

(c) An idle period to service customer transition

(d) A service customer to service customer transition if $x \geq v$

(e) A service customer to service customer transition if $x < v$

Figure 3.2: These plots show the maximum priority process when the different transitions in the MC occurs.
3.2 Derivation of the limiting distributions of the one class problem

\[
\begin{align*}
\frac{\mu}{\lambda + \mu} + \int_0^\infty e^{-\mu x} \frac{\lambda \mu}{\lambda + \mu} dx &= \frac{\mu}{\lambda + \mu} + \frac{\lambda}{\lambda + \mu} \\
&= 1.
\end{align*}
\]

So, the transition probability and density from the Idle state integrate as they should. The left hand side of Expression (3.7) is equal to

\[
\begin{align*}
e^{-\lambda v} \frac{\mu}{\lambda + \mu} + \int_0^v e^{-\lambda(v-x)} \frac{\lambda \mu}{\lambda + \mu} dx &+ \int_v^\infty e^{-\mu(x-v)} \frac{\lambda \mu}{\lambda + \mu} dx \\
&= e^{-\lambda v} \frac{\mu}{\lambda + \mu} + \frac{\mu}{\lambda + \mu} - e^{-\lambda v} \frac{\mu}{\lambda + \mu} + \frac{\lambda}{\lambda + \mu} \\
&= \frac{\mu}{\lambda + \mu} + \frac{\lambda}{\lambda + \mu} \\
&= 1.
\end{align*}
\]

Thus, the transition probabilities from state \( v \) also integrate correctly.

We have now the expressions of the transition probability and density functions. With these we can set up expressions for the limiting distributions of the embedded Markov Chain derived from the maximum priority process in the one class \( M/M/1 \) accumulating priority queue by embedding at points were customers move into service. The expressions for the limiting densities are given by the expressions

\[
\begin{align*}
\pi(0) &= \pi(0)p_{0 \rightarrow 0} + \int_0^\infty \pi(v)p_{v \rightarrow 0} dv, \\
\pi(x) &= \pi(0)f_{0 \rightarrow x} + \int_0^x \pi(v)f_{v \rightarrow x} dv + \int_x^\infty \pi(v)f_{v \rightarrow x, x > v} dv.
\end{align*}
\]

It is known that the stationary probability for the system being in an idle period is equal to \( 1 - \rho \) (see Chapter 2). For the limiting distribution \( \pi(x) \), which is given in Expression (3.9), we know that the accumulated priority \( (x) \) is strictly greater than 0. The priority of a customer can only accumulate when that specific customer is present in the system. In that case the system is in a busy period.

3.2 Derivation of the limiting distributions of the one class problem

To get more insight (which can be used in the two class accumulating priority queue) in the one class accumulating priority queue, we tried to find \( \pi(v) \) with Expressions (3.8) and (3.9).
But we did not succeed by solving the expressions in the time (priority) domain, due to the complexity of the emerging expressions. The problem is the “unknown” function in the integrals ($\pi(v)$). Since it is rather difficult to work out the expressions for the limiting distributions in the time domain, without knowing the exact expression for $\pi(v)$, we apply a new method for solving the expressions. This method makes use of the Laplace transform [4] which is given by

$$\tilde{F}(s) = \mathcal{L}\{f(t)\} = \int_0^\infty e^{-st} f(t) dt$$

and the convolution theorem [4]

$$F_1(s)F_2(s) = \mathcal{L}\{f_1(t) * f_2(t)\} = \int_0^\infty e^{-st} \int_0^t f_1(t-\tau)f_2(\tau)d\tau dt.$$  (3.11)

The idea is to transform the left-hand and right-hand side of the expressions for the limiting distributions from the time domain to the Laplace domain. From there we should be able to solve the expressions to obtain the limiting distributions.

The second expression of the limiting distributions (Expression (3.9)) is

$$\pi(x) = \pi(0)e^{-\mu x} + \int_0^x \pi(v)e^{-\mu(x-v)}dv + \int_x^\infty \pi(v)e^{-\lambda(v-x)}dv.$$  (3.12)

It can be seen that every term on the right-hand side of the expression has the same constant factor $\frac{\lambda\mu}{\lambda+\mu}$. Furthermore, we recognize that the second term of the right-hand side is a convolution, in Appendix A the proof of the convolution theorem is given. We can rewrite the third term as the difference of two other terms, one of them an integral over the interval $[0, \infty)$ and the other, an integral over the interval $[0, x]$, in this term we recognize a convolution again. So, we can rewrite Expression (3.12) as

$$\pi(x) = \frac{\lambda\mu}{\lambda+\mu} \left( \pi(0)e^{-\mu x} + \int_0^x \pi(v)e^{-\mu(x-v)}dv + \int_0^\infty \pi(v)e^{-\lambda(v-x)}dv - \int_0^x \pi(v)e^{\lambda(x-v)}dv \right).$$  (3.13)

We move the left hand side of Expression (3.13) to the right hand side and take the Laplace transform (definition given in Expression (3.10)) and we obtain

$$0 = -\int_0^\infty \pi(x)e^{-sx}dx + \frac{\lambda\mu}{\lambda+\mu} \left( \int_0^\infty \pi(0)e^{-\mu x}e^{-sx}dx + \int_0^\infty \int_0^x \pi(v)e^{-\mu(x-v)}dv e^{-sx}dx + \int_0^\infty \int_0^\infty \pi(v)e^{-\lambda(v-x)}dv e^{-sx}dx - \int_0^\infty \int_0^x \pi(v)e^{\lambda(x-v)}dv e^{-sx}dx \right).$$  (3.14)

We work out the right hand side of Expression (3.14) and we see that
3.2. Derivation of the limiting distributions of the one class problem

\[-\int _{0}^{\infty }\pi (x)e^{-sx}dx + \frac{\lambda \mu }{\lambda + \mu }\left( \int _{0}^{\infty }\pi (0)e^{-\mu x}e^{-sx}dx + \int _{0}^{\infty }\int _{0}^{x}\pi (v)e^{-\mu (x-v)}dv e^{-sx}dx + \right)\]

\[\int _{0}^{\infty }\int _{0}^{\infty }\pi (v) e^{-\lambda (v-z)}dve^{-sz}dx - \int _{0}^{\infty }\int _{0}^{z}\pi (v)e^{\lambda (z-v)}dve^{-sz}dx\]

\[= - \tilde{\pi}(s) + \frac{\lambda \mu }{\lambda + \mu } \left( \frac{\pi (0)}{s + \mu } + \tilde{\pi}(s) \frac{1}{s + \mu } + \tilde{\pi}(\lambda) \frac{1}{s - \lambda} - \tilde{\pi}(s) \frac{1}{s - \lambda}\right)\]

\[= \frac{\lambda \mu (s - \lambda) \pi (0) + \lambda \mu (s + \mu) \tilde{\pi}(\lambda) - s(s - \lambda + \mu)(s + \mu)\tilde{\pi}(s)}{(s - \lambda)(s + \mu)(\lambda + \mu)}\]

(3.15)

The expression that we obtain should be equal to the left hand side of Expression (3.14), i.e. equal to 0. The Laplace transform that is obtained must be analytic within the right half plane where \(Re(s) \geq 0\). We observe a root in the denominator which is situated in this plane, i.e. \(s = \lambda\). So, the numerator has to be equal to zero at \(s = \lambda\). When this is substituted in the numerator it is observed that this is true. However, we do not obtain the expressions for \(\tilde{\pi}(s)\) and \(\tilde{\pi}(\lambda)\).

Therefore, we go back to Expression (3.14) and solve the integrals, substitute \((1 - \rho)\) for \(\pi(0)\), bring all the terms which include the function \(\tilde{\pi}(s)\) to the left-hand side and the constant term \(\tilde{\pi}(\lambda)\) to the right-hand side of the expression. Doing this we obtain

\[\tilde{\pi}(s) \left( \frac{(\lambda + \mu)(s + \mu)(s - \lambda) - \lambda\mu(s - \lambda) + \lambda\mu(s + \mu)}{\lambda\mu(s + \mu)(s - \lambda)} \right) = \frac{\lambda(\mu - \lambda)(s - \lambda) + \tilde{\pi}(\lambda)\lambda\mu(s + \mu)}{\lambda\mu(s + \mu)(s - \lambda)}\]

and provided the denominator of this expression is non-zero, \(\tilde{\pi}(s)\) is given by

\[\tilde{\pi}(s) = \frac{\lambda(\mu - \lambda)(s - \lambda) + \tilde{\pi}(\lambda)\lambda\mu(s + \mu)}{(\lambda + \mu)(s + \mu)(s - \lambda) - \lambda\mu(s - \lambda) + \lambda\mu(s + \mu)}\]

(3.16)

In this expression we have two unknowns, i.e. the function \(\tilde{\pi}(s)\) and the constant \(\tilde{\pi}(\lambda)\).

3.2.1 Derivation of \(\tilde{\pi}(\lambda)\)

The Laplace transform must be analytic within the right half plane where \(Re(s) \geq 0\). If we evaluate Expression (3.16) we observe that the transform converges in the region where \(Re(s) \geq 0\). Furthermore, if we observe the denominator of the Laplace transform we can see that it contains a quadratic expression

\[(\lambda + \mu)(s + \mu)(s - \lambda) - \lambda\mu(s - \lambda) + \lambda\mu(s + \mu),\]

which can be simplified to

\[(\lambda + \mu)(s^2 + (\mu - \lambda)s),\]
from which we directly obtain the roots \( s = 0 \) and \( s = \lambda - \mu \). The second root is located in the left-half plane since \( \mu > \lambda \), since we only consider a stable system. Since the Laplace transform is analytic for \( \text{Re}(s) \geq 0 \), we know that the numerator of Expression (3.16) should have the same root as the denominator. Hence the expression that we need to solve, by substitution of \( s = 0 \), for obtaining \( \tilde{\pi}(\lambda) \) is given by

\[
\lambda (\mu - \lambda)(-\lambda) + \tilde{\pi}(\lambda) \lambda \mu (\mu) = 0.
\]

Solving for \( \tilde{\pi}(\lambda) \) gives

\[
\tilde{\pi}(\lambda) = \frac{\lambda^2 (\mu - \lambda)}{\lambda \mu^2} = \frac{\lambda (\mu - \lambda)}{\mu} = \frac{\mu - \lambda}{\mu}.
\]

So, the constant \( \tilde{\pi}(\lambda) \) is obtained.

### 3.2.2 Derivation of \( \tilde{\pi}(s) \)

The expression for \( \tilde{\pi}(s) \) is given by Expression (3.16). After the derivation of \( \tilde{\pi}(\lambda) \) we know several things. The first is the value for \( \tilde{\pi}(\lambda) \) itself, moreover we know also that the roots for the denominator of \( \tilde{\pi}(s) \) are \( s = 0 \) and \( s = \lambda - \mu \). We also know that the root of the numerator is \( s = 0 \).

Substituting the expression for \( \tilde{\pi}(\lambda) \) in the numerator of Expression (3.16)

\[
\lambda (\mu - \lambda)(s - \lambda) + \rho \frac{\mu - \lambda}{\mu} \lambda \mu (s + \mu) = (\mu - \lambda)(\lambda(s - \lambda) + \rho \lambda(s + \mu)),
\]

which we simplify further to

\[
(\mu - \lambda)(\lambda s + \rho \lambda s) = (\lambda \mu - \rho \lambda^2) s,
\]

and taking \( \rho \) outside the brackets

\[
\rho (\mu^2 - \lambda^2) s = \rho (\mu + \lambda)(\mu - \lambda) s.
\]

Now, we have an expression for the numerator of \( \tilde{\pi}(s) \) in which we recognize the root \( s = 0 \).

So, we obtain the expression for \( \tilde{\pi}(s) \), which is given by

\[
\tilde{\pi}(s) = \frac{\rho (\mu + \lambda)(\mu - \lambda) s}{(\lambda + \mu)(s + \mu - \lambda) s} = \frac{\mu - \lambda}{s + \mu - \lambda}.
\]  

(3.17)

This is the stationary waiting time distribution of the one class maximum priority process in the M/M/1 accumulating priority queue in the Laplace domain. Since only one customer class is considered, the accumulated priority density function is equal to the waiting time probability
density function in the $M/M/1$ queue. The waiting time distribution in an $M/M/1$ queue in
the time domain is given by [2]

\[ w(t) = (\mu - \lambda)e^{-(\mu - \lambda)t}. \]  

(3.18)

From the previous chapter it is known that the virtual workload process is connected to the
maximum priority process. The accumulated priority density function is a function of priority
instead of time. However, since we have $b = 1$ we can substitute the accumulated priority
$x \in (0, \infty)$ for the time $t$ in Expression (3.18). So, for the limiting density as function of the
accumulated priority we obtain

\[ \pi(x) = \rho(\mu - \lambda)e^{-(\mu - \lambda)x}. \]  

(3.19)

If we transform this expression to the Laplace domain we obtain

\[ \int_0^\infty e^{-sx} \rho(\mu - \lambda)e^{-(\mu - \lambda)x} dx = \rho \frac{\mu - \lambda}{s + \mu - \lambda}. \]  

(3.20)

Which is the same result as obtained in Expression (3.17).

Now, we recall the expressions worked out in Expression (3.15). With those expressions we
want to check whether the right hand side of Expression (3.14) equals its right hand side, i.e.
equals $0$. When we substitute the obtained function $\tilde{\pi}(s)$ and the constants $\pi(0)$ and $\tilde{\pi}(\lambda)$ in
this expression we obtain

\[
\frac{\lambda \mu (s - \lambda) \pi(0) + \lambda \mu (s + \mu) \tilde{\pi}(\lambda) - s(s - \lambda + \mu)(\lambda + \mu)\pi(s)}{\lambda \mu \left(\frac{(s-\lambda)(\mu-\lambda)}{\mu} + \frac{(s+\mu)(\mu-\lambda)}{\mu^2}\right) - \frac{s\lambda(\mu-\lambda)(\lambda+\mu)}{\mu}}
\]

\[
= \frac{\lambda \mu (s - \lambda)(\mu - \lambda) + \lambda(s + \mu)(\mu - \lambda) - s\lambda \mu (\mu - \lambda)(\lambda + \mu)}{(s - \lambda)(s + \mu)(\lambda + \mu)\mu^2}
\]

\[
= \frac{\lambda \mu (\mu^2 s - \lambda^2 s) - s\lambda \mu (\mu^2 - \lambda^2)}{(s - \lambda)(s + \mu)(\lambda + \mu)\mu^2}
\]

\[
= 0.
\]

Thus, with working out the expressions with the obtained function $\tilde{\pi}(s)$ and constant $\tilde{\pi}(\lambda)$, we
proved that the right hand side of Expression (3.14) is equal to the left hand side, i.e. $0$.

So, with this method we are able to solve the expressions for the limiting distributions in the
Laplace domain, which is also a new way to derive the waiting time distribution of the $M/M/1$
queue. With this method we want to derive the stationary waiting time distribution for the
two class maximum priority process in the $M/M/1$ accumulating priority queue.
Chapter 3. Study of the one class maximum priority process in the $M/M/1$ queue
Chapter 4

Study of the two class maximum priority process in the $M/M/1$ queue

In the previous chapter we provided a method for solving expressions for the limiting distributions of the one class accumulating priority queue. We are able to derive the stationary waiting time distribution of the $M/M/1$ queue. In this chapter we consider the maximum priority process in an $M/M/1$ queueing system with two customer classes. In the analysis of this system, $\lambda_1$ and $\lambda_2$ denote the Poisson arrival rates of class 1 and class 2 customers, respectively, and $\mu$ denotes the service rate of the system. The utilization is denoted with $\rho = \frac{\lambda_1 + \lambda_2}{\mu}$. The priority rates of the different customer classes are denoted by $b_1$ and $b_2$, respectively. For the ease of notation of the expressions in this chapter we set $b_1 = 1$ and $b_2 < 1$. So, as was the case with priority rate $b$ in the previous chapter (Chapter 3) the reader will not find the parameter $b_1$ in the expressions in this chapter.

This chapter contains four sections. In the first section we define the transition probabilities and densities of a Markov Chain nested for the maximum priority process at the times at which customers enter service. With the defined transition probabilities and densities we are able to set up the expressions for the limiting distribution of this Markov model, we derive these expressions in the second section. We use the Laplace transform to obtain the limiting distributions in the Laplace domain in the third section. In the fourth and last section we try to derive the stationary waiting time distribution of the two class maximum priority process in the $M/M/1$ queue.

4.1 Markov Chain of the two class maximum priority process

The maximum priority process in the two class accumulating priority queue is a continuous-time process. As was the case in the one class accumulating priority queue, see Chapter 3, we only
study the discrete time process \((M^n_1, M^n_2)\) that we derive from the continuous-time process by only taking the embedded points where customers enter service into consideration. This process is depicted schematically in Figure 4.1 in a Markov Chain (MC). The MC of this Markov model contains; (i) the state at which the server is idle (Idle Period), (ii) a 1-D continuum set of states at which an unaccredited customer enters service (Service Unaccredited Customer) and (iii) a 2-D continuum set of states at which service of an accredited customer commences (Service Accredited Customer). In the figure these two continuums, the Service Accredited Customer and Service Unaccredited Customer boxes, are encircled by a red dashed line, which also indicates that the Markov model is in a busy period when it is in one of the encircled sets.

As mentioned earlier, in Section 2.4, we make a distinction between two types of customers, the accredited and unaccredited customers. Accredited customers must be from class 1 and enter service with an accumulated priority that is in the priority interval \([M_2(t), M_1(t)]\), that is with a higher priority than any class 2 customer could possibly have at that time. Unaccredited customers enter service with an accumulated priority that lies in the priority interval \([0, M_2(t)]\); these customers are a mixture of class 1 and class 2 customers.

![Figure 4.1: The Markov Chain of the maximum priority process with two different customer classes.](image)

The continuous-state Markov process has transition probabilities into the Idle Period state \((0,0)\), a 1-dimensional transition density into the Service Unaccredited Customer set \((x,x)\), with accumulated priority \(x \in (0, \infty)\), and a 2-dimensional transition density into the Service Accredited Customer set \((x,y)\), with accumulated priorities \(x \in (0, \infty)\) and \(y \in (b_2 x, x)\) from any state present in the MC, see Figure 4.1. In total we distinguish nine different types of transitions in this continuous-state Markov process, i.e. \((0,0) \rightarrow (0,0), (0,0) \rightarrow (x,x), (0,0) \rightarrow (x,y), (v,v) \rightarrow (0,0), (v,v) \rightarrow (x,x), (v,v) \rightarrow (x,y), (v,w) \rightarrow (0,0), (v,w) \rightarrow (x,x)\) and \((v,w) \rightarrow (x,y)\), in which \(v \in (0, \infty), w \in (b_2 v, v), x \in (0, \infty)\) and \(y \in (b_2 x, x)\) denote the accumulated priorities. The transition probabilities and densities of the MC are

1. **Idle to Idle:** When this transition occurs we know that the single service of duration \(u \in (0, \infty)\) received by a customer who arrives to an empty queue finishes before any arrival
of a new customer (accredited or unaccredited) occurs. So, the accumulated priorities at the moment the service finishes are equal to \((0, 0)\). A schematic representation of this transition is depicted in Figure 4.2a. By conditioning on \(u\), the transition probability is given by

\[
P((M^{n+1}_1, M^{n+1}_2) = (0, 0) | (M^n_1, M^n_2) = (0, 0))
= \int_0^\infty \mu e^{-\mu u} e^{-(\lambda_1 + \lambda_2)u} du
= \frac{\mu}{\mu + \lambda_1 + \lambda_2}.
\] (4.1)

In Expression (4.1) we recognize again the “race” between the service and arrival processes, in which in this case the service wins. For the ease of notation throughout the rest of this report we shall refer to this expression as \(p_{0,0\rightarrow0,0}\).

2. Idle to Accredited: If \((M^n_1, M^n_2) = (0, 0)\) and the value of \((M^{n+1}_1, M^{n+1}_2)\) is equal to \((x, y)\) with \(x > y > 0\), then of necessity the service must have lasted for a time \(\frac{y}{\lambda_2}\). The accredited customer commencing service at time point \(n + 1\) must have arrived a time \(\frac{y}{\lambda_2} - x\) after the start of the service of the customer who initiated the busy period. A schematic representation of the transition is depicted in Figure 4.2b. So for \(y > 0\) and \(x \in (y, \frac{y}{\lambda_2})\) the following transition probability density function holds

\[
P((M^{n+1}_1, M^{n+1}_2) = (x, y) | (M^n_1, M^n_2) = (0, 0))
= \frac{1}{b_2} \mu e^{-\mu \frac{y}{\lambda_2}} \lambda_1 e^{-\lambda_1 (\frac{y}{\lambda_2} - x)}
= \frac{1}{b_2} \mu \lambda_1 e^{\lambda_1 x} e^{-(\mu + \lambda_1) \frac{y}{\lambda_2}}.
\] (4.2)

Throughout the rest of this report we shall denote this expression with \(f_{0,0\rightarrow x,y}\).

3. Idle to Unaccredited: If \((M^n_1, M^n_2) = (0, 0)\) and the value of \((M^{n+1}_1, M^{n+1}_2)\) is \((x, x)\) with \(x > 0\), then we know that the service lasted for a time \(u > \frac{x}{\lambda_2}\), the arrived customer is an unaccredited customer who is either from class 1 or class 2. If this customer is a class 1 customer we know that he arrived at time \(u - x\) after the start of the service of the customer who initiated the busy period, with no class 2 customer arriving in \([0, u - \frac{x}{\lambda_2}]\). When the unaccredited customer is a class 2 customer we know that he arrived at time \(u - \frac{x}{\lambda_2}\) after the start of the service of the customer who initiated the busy period and no class 1 customer arriving in \([0, u - \frac{x}{\lambda_2}]\). A schematic representation of this transition is depicted in Figure 4.2c. So, for \(x > 0\) and by conditioning on \(u\), the transition probability density function is given by
Chapter 4. Study of the two class maximum priority process in the M/M/1 queue

\[ P((M_1^{n+1}, M_2^{n+1}) = (x, x)| (M_1^n, M_2^n) = (0, 0)) \]

\[ = \int_0^\infty \mu e^{-\mu u} e^{-\lambda_1 (1 - b_2) u} (\lambda_1 + \frac{\lambda_2}{b_2}) e^{-(\lambda_1 + \frac{\lambda_2}{b_2}) (b_2 u - x)} du \]

\[ = \frac{e^{-(\mu + \lambda_1 + \lambda_2) x}}{\mu} \int_0^\infty e^{-(\mu + \lambda_1 + \lambda_2) u} du \]

Throughout the rest of this report we shall denote Expression (4.3) with \( f_{0,0 \rightarrow x,x} \).

![Priority Diagrams](image)  

(a) An idle to idle transition  
(b) An idle to accredited customer transition  
(c) An idle to unaccredited customer transition

Figure 4.2: These plots show the maximum priority process when the different transitions from the idle state in the MC occurs.

4. **Accredited to Idle**: When this transition occurs in the MC we know that there is no new arrival of a customer (either accredited or unaccredited) at the system before the server finishes its service. In this case the server finishes service, with service time \( u \in (0, \infty) \), of
an accredited customer. Due to the fact that no new customer arrives at the system, we
know there is no class 1 customer, who becomes accredited, that arrived in the interval
\([w + b_2 u, v + u]\). There is also no arrival of an unaccredited customer with rate \(\lambda_1 + \frac{\lambda}{b_2}\) in
the interval \([0, w + b_2 u]\). So, the accumulated priorities at the moment the service finishes
are equal to \((0, 0)\). A schematic representation of this transition is depicted in Figure
4.3a. By conditioning on the service \(u\), the transition probability function is given by

\[
P(\{M_1^{n+1}, M_2^{n+1}\} = (0, 0)|\{M_1^n, M_2^n\} = (v, w))
= \int_0^\infty \mu e^{-\mu u} e^{-\lambda_1((v+u)-(w+b_2 u))} e^{-(\lambda_1+\frac{\lambda}{b_2})(w+b_2 u)}
= \int_0^\infty \mu e^{-(\mu+\lambda_1(1-b_2)+\lambda_1 b_2 + \lambda_2) u} e^{-\lambda_1(v-w)} e^{-(\lambda_1+\frac{\lambda}{b_2})w} du
= e^{-\lambda_1 v} e^{-\frac{\lambda}{b_2} w} \frac{\mu}{\mu + \lambda_1 + \lambda_2}.
\]  

(4.4)

Throughout the rest of this report we shall denote Expression (4.4) with \(p_{v,w\to 0,0}\).

5. **Accredited to Accredited:** If \((M_1^n, M_2^n) = (v, w)\) and the value of \((M_1^{n+1}, M_2^{n+1})\)
is equal to \((x, y)\) with \(x > y > 0\), then of necessity the service must have lasted for a
time \(\frac{w-y}{b_2}\). The accredited customer commencing service at time point \(n + 1\) must have
arrived a time \(\frac{w-y}{b_2} - x\) after the start of the service of the previous customer who was
an accredited customer when he commenced service. A schematic representation of the
transition is depicted in Figure 4.3b. So for \(y \in (w, \infty)\) and \(x \in (y, v + \frac{w-y}{b_2})\) the following
transition probability density function holds

\[
P(\{M_1^{n+1}, M_2^{n+1}\} = (x, y)|\{M_1^n, M_2^n\} = (v, w))
= \frac{1}{b_2} e^{-\mu \frac{v-w}{b_2}} \lambda_1 e^{-\lambda_1(v+\frac{w-y}{b_2}-x)}
= \frac{1}{b_2} e^{-\lambda_1(v-x)} e^{-(\mu+\lambda_1)\frac{w-y}{b_2}}.
\]  

(4.5)

Throughout the rest of this report we shall denote this expression with \(f_{v,w\to x,y}\).

6. **Accredited to Unaccredited:** If \((M_1^n, M_2^n) = (v, w)\) and the value of \((M_1^{n+1}, M_2^{n+1})\)
is \((x, x)\) with \(x > 0\), then we know that the service lasted for a time \(u > \frac{w-y}{b_2}\), the
arrived customer is an unaccredited customer who is either from class 1 or class 2. So,
we know there is no arrival of an class 1 customer with rate \(\lambda_1\) in the interval \([w + b_2 u, v + u]\) who becomes accredited during waiting. We know that there is an arrival of a
customer with arrival rate \(\lambda_1 + \frac{\lambda}{b_2}\) in the interval \([0, w + b_2 u]\). Schematic representations of
these transitions are depicted in Figures 4.3c and 4.3d. As can be seen in these plots,
a distinction is made between two possibilities, i.e. \(x < w\) and \(x \geq w\).

(a) when \(x < w\), by conditioning on the length of the service time \(u\) and for \(v \in (0, \infty)\),
\(w \in (b_2 v, v)\) and \(x \in (0, w)\) the probability density function is given by

\[
\text{Conditional Pr}(X = x|W = w, V = v) =\]
Chapter 4. Study of the two class maximum priority process in the M/M/1 queue

\[ P(M_1^{n+1}, M_2^{n+1}) = (x, x) | (M_1^n, M_2^n) = (v, w), x < w \]
\[ = \int_0^\infty \mu e^{-\mu u} e^{-\lambda_1(v-w+(1-b_2)u)}(\lambda_1 + \frac{\lambda_2}{b_2})e^{-(\lambda_1 + \frac{\lambda_2}{b_2})(w+b_2u-x)} du \]
\[ = \int_0^\infty \mu(\lambda_1 + \frac{\lambda_2}{b_2})e^{-(\mu+\lambda_1(1-b_2)+\lambda_2b_2+\lambda_2)u}e^{-\lambda_1(v-w)}e^{-(\lambda_1 + \frac{\lambda_2}{b_2})(w-x)} du \]
\[ = e^{-\lambda_1(v-w)}(\lambda_1 + \frac{\lambda_2}{b_2}) \mu(\lambda_1 + \frac{\lambda_2}{b_2}) \]

Throughout the rest of this report we shall denote this expression with \( f_{v,w \rightarrow x,x|x < w} \).

(b) when \( x \geq w \), by conditioning on the length of the service time \( u \) and for \( v \in (0, \infty) \), \( w \in (b_2v, v) \) and \( x \in (w, \infty) \) the probability density function is given by

\[ P(M_1^{n+1}, M_2^{n+1}) = (x, x) | (M_1^n, M_2^n) = (v, w), x \geq w \]
\[ = \int_{v+w}^\infty \mu e^{-\mu u} e^{-\lambda_1(v-w+(1-b_2)u)}(\lambda_1 + \frac{\lambda_2}{b_2})e^{-(\lambda_1 + \frac{\lambda_2}{b_2})(w+b_2u-x)} du \]
\[ = \int_0^\infty \mu e^{-(\mu+\lambda_1(1-b_2)+\lambda_2b_2+\lambda_2)u}e^{-\lambda_1(v-w)}e^{-(\lambda_1 + \frac{\lambda_2}{b_2})(w-x)} du \]
\[ = e^{-\lambda_1(v-w)}(-\mu+\lambda_1(1-b_2)+\lambda_2b_2) \mu(\lambda_1 + \frac{\lambda_2}{b_2}) \]

Throughout the rest of this report we shall denote this expression with \( f_{v,w \rightarrow x,x|x \geq w} \).

7. Unaccredited to Idle: When this transition occurs in the MC we know that there is no new arrival of a customer (either accredited or unaccredited) at the system before the service, \( u \in (0, \infty) \), of an unaccredited customer finishes. So, we know there is no class 1 customer, which becomes accredited, in the interval \((v+b_2u, v+u)\). There is also no arrival of an unaccredited customer with rate \( \lambda_1 + \frac{\lambda_2}{b_2} \) in the interval \((0, v+b_2u)\). So, the accumulated priorities at the moment the service finishes are equal to \( (0, 0) \). A schematic representation of this transition is depicted in Figure 4.4a. By conditioning on the service \( u \), the transition probability function is given by

\[ P(M_1^{n+1}, M_2^{n+1}) = (0, 0) | (M_1^n, M_2^n) = (v, v) \]
\[ = \int_0^\infty \mu e^{-\mu u} e^{-\lambda_1(v-b_2u)}e^{-(\lambda_1 + \frac{\lambda_2}{b_2})(v+b_2u)} du \]
\[ = \int_0^\infty \mu e^{-(\mu+\lambda_1(1-b_2)+\lambda_2b_2+\lambda_2)u}e^{-(\lambda_1 + \frac{\lambda_2}{b_2})v} du \]
\[ = e^{-(\lambda_1 + \frac{\lambda_2}{b_2})v} \mu(\lambda_1 + \frac{\lambda_2}{b_2}) \]

Throughout the rest of this report we shall denote this expression with \( p_{v,v \rightarrow 0,0} \).
4.1. Markov Chain of the two class maximum priority process

Figure 4.3: These plots show the maximum priority process when the different transitions from the accredited service set of states in the MC occurs.

8. Unaccredited to Accredited: If \((M^n_1, M^n_2) = (v, v)\) and the value of \((M^{n+1}_1, M^{n+1}_2)\) is equal to \((x, y)\) with \(x > y > 0\), then of necessity the service must have lasted for a time \(\frac{y-v}{b_2}\). The accredited customer commencing service at time point \(n + 1\) must have arrived a time \(\frac{y-v}{b_2} - x\) after the start of the service of the previous customer who was an unaccredited customer when he commenced service. A schematic representation of the transition is depicted in Figure 4.4b. So for \(y \in (v, \infty)\) and \(x \in (y, v + \frac{y-v}{b_2})\) the following transition probability density function holds

\[
P\left((M^{n+1}_1, M^{n+1}_2) = (x, y) | (M^n_1, M^n_2) = (v, v)\right) = \frac{1}{b_2} \mu e^{-\frac{\mu}{b_2} y} \lambda_1 e^{-\lambda_1 (v + \frac{y-v}{b_2} - x)}
\]

\[
= \frac{1}{b_2} \mu \lambda_1 e^{-\lambda_1 (v - x)} e^{-\left(\mu + \lambda_1\right) \left(\frac{y-v}{b_2}\right)}.
\]

(4.9)

Throughout the rest of this report we shall denote this expression with \(f_{v,v\rightarrow x,y}\).
9. **Unaccredited to Unaccredited:** If \((M_1^n, M_2^n) = (v, v)\) and the value of \((M_1^{n+1}, M_2^{n+1})\) is \((x, x)\) with \(x > 0\), then we know that the service lasted for a time \(u > \frac{x}{b_2}\), the arrived customer is an unaccredited customer who is either from class 1 or class 2. So, we know there is no arrival of an unaccredited customer who is either from class 1 or class 2. Hence, we expect that

\[
P_{\text{Unaccredited to Unaccredited}} = \begin{cases} (x, x) & x > 0, \\ (v, v) & x = 0. \end{cases}
\]

As before in Chapter 3, the total transition probability out of every state of the MC should be equal to one. Thus we expect that

\[
p_{0,0|0,0} + \int_{0}^{\infty} \int_{y}^{2} f_{0,0|x,y} dx dy + \int_{0}^{\infty} f_{0,0|x,\infty} dx = 1, \tag{4.12}
\]

\[
p_{v,v|0,0} + \int_{v}^{w} \int_{y}^{w-x} f_{v,v|x,y} dx dy + \int_{v}^{w} f_{v,v|x,\infty} dx + \int_{w}^{\infty} f_{v,v|x|x<x} dx + \int_{w}^{\infty} f_{v,v|x|x\geq w} dx = 1 \tag{4.13}
\]
4.1. Markov Chain of the two class maximum priority process

Figure 4.4: These plots show the maximum priority process when the different transitions from the unaccredited service set of states in the MC occurs.

\[
\begin{align*}
\text{(a) Unaccredited customer to idle transition} & \quad \text{(b) Unaccredited customer to accredited customer transition} \\
\text{(c) Unaccredited customer to unaccredited customer transition, } x < v & \quad \text{(d) Unaccredited customer to unaccredited customer transition, } x \geq v
\end{align*}
\]

The left hand side of every expression is evaluated and they are all equal to one, so the transition probabilities from every state integrate as they should. The full evaluation of Expression (4.12), (4.13) and (4.14) can be found in Appendix B.
4.2 Expressions for the limiting distributions of the two class maximum priority process

In the previous section one has seen the derivation of the transition probability density functions belonging to the transitions as presented in the MC in Figure 4.1. In our search of the stationary waiting time distribution of the maximum priority process in the two class \(M/M/1\) queue we need to define the expressions for the limiting distributions the same way we did in the study of the maximum priority process in the one class \(M/M/1\) system. These expressions for the limiting distributions should have the following form

\[
\pi(0,0) = \pi(0,0)p_{0,0\rightarrow 0,0} + \int_0^\infty \pi(v,v)p_{v,v\rightarrow 0,0}dv + \int_0^\infty \int_{v_2}^{w_2} \pi(v,w)p_{v,w\rightarrow 0,0}dvdw, \quad (4.15)
\]

\[
\pi(x,x) = \pi(0,0)f_{0,0\rightarrow x,x} + \int_0^x \pi(v,v)f_{v,v\rightarrow x,x\mid x\geq y}dv + \int_x^\infty \pi(v,v)f_{v,v\rightarrow x,x\mid x\leq y}dv + \int_0^x \int_w^{w_2} \pi(v,w)f_{v,w\rightarrow x,x\mid x\geq w}dvdw + \int_x^\infty \int_w^{w_2} \pi(v,w)f_{v,w\rightarrow x,x\mid x\leq w}dvdw, \quad (4.16)
\]

and

\[
\pi(x,y) = \pi(0,0)f_{0,0\rightarrow x,y} + \int_0^{\frac{x-y}{b_2}} \pi(v,v)f_{v,v\rightarrow x,y}dv + \int_{\frac{x-y}{b_2}}^y \int_w^{w_2} \pi(v,w)f_{v,w\rightarrow x,y}dvdw + \frac{y-b_2 x}{1-b_2^2} \int_{\frac{x-y}{b_2} + x}^{w_2} \pi(v,w)f_{v,w\rightarrow x,y}dvdw.
\]

(4.17)

The integral bounds of the expressions follow from observation of the domain for which the integrals hold in the \((M_1(t), M_2(t))\) plane. A schematic representation of this plane can be found in Figure 4.5a. If we take this figure into consideration we see that it follows from our choice of \(v \in (0, \infty)\) that \(w \in (b_2 v, v)\).

For the integral bounds of the third expression we take a closer look at the \((M_1(t), M_2(t))\) plane, see Figure 4.5b. If we take the transition from unaccredited to accredited into consideration. Thus from an accumulated priority \(v\) to accumulated priorities \((x, y)\). The transition originates from a point on the line \(M_1(t) = M_2(t)\) with lower bound of 0 and an upper bound of the point on the line, which is \(\frac{y-b_2 x}{1-b_2^2}\) (the red line in the figure). For a transition from accredited to accredited we know that the point \((x, y)\) can be reached from a point \((v, w)\) located in the areas I and II in the wedge, see Figure 4.5b. Therefore, \(f_{v,w\rightarrow x,y}\) is integrated over two distinct domains. For area I we integrate over \(w \leq v \leq \frac{w-y}{b_2}\) and \(0 \leq w \leq \frac{y-b_2 x}{1-b_2^2}\) and for area II over \(\frac{w-y}{b_2} + x \leq v \leq \frac{w}{b_2}\) and \(\frac{y-b_2 x}{1-b_2^2} \leq w \leq y\).
4.2. Expressions for the limiting distributions of the two class maximum priority process

Figure 4.5: Plot of $M_2(t)$ versus $M_1(t)$ in 4.5a and with the transition bounds for the third expression in 4.5b, transitions to the point $(x, y)$ are only possible form the origin (idle), red line (unaccredited) or areas I and II (accredited).

Note that we made the choice of integrating the terms with $\pi(v, w)$ in Expression (4.16) over variable $v$ before integrating over $w$. We obtain the integral limits by examination of the $(M_2(t), M_1(t)$ (or $(v, w))$ plot, see Figure 4.5a. The reason we choose for these integral limits is because when we use these limits we have less terms in the expressions to solve, if we make the choice for the case of first integrating over $w$ and then over $v$ we get for the fourth and fifth term at the right hand side of Expression (4.16) the following

$$
\int_0^{\frac{x}{b_2}} \int_{b_2 v}^{\min(x, v)} f_{v, w \rightarrow x, x} |x \geq w| dv \, dw + \int_{b_2}^{\infty} \int_{b_2 v}^{x} f_{v, w \rightarrow x, x} |x < w| dv \, dw.
$$

When we write the integrals further out and we get rid of the min and max functions in the integral bounds we obtain

$$
\int_0^{\frac{x}{b_2}} \int_{b_2 v}^{0} f_{v, w \rightarrow x, x} |x \geq w| dv \, dw + \int_{b_2}^{\infty} \int_{b_2 v}^{x} f_{v, w \rightarrow x, x} |x \geq w| dv \, dw + \\
\int_x^{\infty} \int_{b_2 v}^{0} f_{v, w \rightarrow x, x} |x < w| dv \, dw + \int_{b_2}^{\infty} \int_{b_2 v}^{x} f_{v, w \rightarrow x, x} |x < w| dv \, dw.
$$

Now, it is rather easy to see that we have obtained four terms instead of the two terms we obtain by integration over $v$ before integration over $w$. So, we prefer to do the integration as defined in Expression (4.16).

By substitution of the derived densities, see Section 4.1, in the expressions for the limiting distributions ((4.15), (4.16) and (4.17)) we arrive at the expressions we need to solve for obtaining the stationary waiting time distribution of the maximum priority process in the $M/M/1$ queueing system with two customer classes.
Chapter 4. Study of the two class maximum priority process in the $M/M/1$ queue

4.3 Transformation of the expressions for the limiting distributions to the Laplace domain

We want to use the Laplace transforms the same way as we did in the one class accumulating priority queue, see Chapter 3. We know that $\pi(0, 0)$ is a point mass belonging to the idle period of the Markov model. The $\pi(x, x)$ expression is a 1D distribution only dependent on the accumulated priority $x$ and it describes the stationary waiting time distribution of the unaccredited customers. The expression for $\pi(x, y)$ is a 2D distribution dependent on the accumulated priorities $x$ and $y$, and belonging to the waiting time distribution of the accredited customers. If we take a look at the $(M_2(t), M_1(t))$ plot as depicted in Figure 4.5a, these distributions are located as follows; the $\pi(0, 0)$ mass can be found at the origin of the plot, the mass of the $\pi(x, x)$ distribution can be found along the line where $M_2(t) = M_1(t)$ and the mass of the $\pi(x, y)$ distribution can be found inside the wedge between the lines $M_2(t) = b_2 M_1(t)$ and $M_2(t) = M_1(t)$.

The next step in order to obtain the stationary waiting time distribution is to solve the expressions for the limiting distributions in the Laplace domain. Therefore we recall the definition of the Laplace transform given in Equation (3.10). This definition should be sufficient for solving the equation for $\pi(x, x)$ (which is the 1D distribution). However, for the 2D distribution given by $\pi(x, y)$ we need the double Laplace transform, which we define as

$$
\hat{F}(s_1, s_2) = \mathcal{L}_x \mathcal{L}_y \{\pi(x, y)\} = \tilde{\pi}(s_1, s_2) = \int_0^\infty e^{-s_2 y} \int_y^{\infty} e^{-s_1 x} \pi(x, y) dx dy \quad \text{if} \quad (x, y) \in \text{range}.
$$

(4.18)

The range is defined as $y \in (0, \infty)$ and $x \in (y, \frac{y}{b_2})$. If we take a look at the $(M_1(t), M_2(t))$ (or $(x, y)$) plot in Figure 4.5a we recognize that the function is equal to zero outside the domain, since no mass is present in those regions. So, we have a valid double Laplace transform.

In this section the reader will find three subsections. In these subsections the transformations to the Laplace domain of respectively the Expressions (4.15), (4.16) and (4.17) are worked out.

4.3.1 Transformation of the expression for $\pi(0, 0)$

When we evaluate the expression for the limiting distribution of $\pi(0, 0)$, given in Expression (4.15), we observe that there are two terms with integrals to be solved. We evaluate these terms in the integrals and for the right hand side of Expression (4.15) we directly recognize the definition of the single Laplace transform in the second term. In the third term we observe the double Laplace transform as defined in Expression (4.18), so for the right hand side of Expression (4.15) we obtain

$$
\frac{\mu}{\mu + \lambda_1 + \lambda_2} \left( \pi(0, 0) + \int_0^\infty \pi(v, v) e^{-\left(\lambda_1 + \frac{\lambda_2}{b_2}\right)v} dv + \int_0^\infty \int_0^{\frac{v}{b_2}} \pi(v, w) e^{-\lambda_1 v} e^{-\frac{\lambda_2}{b_2} w} dw dv \right)
$$

$$
= \frac{\mu}{\mu + \lambda_1 + \lambda_2} \left( \pi(0, 0) + \tilde{\pi} \left( \lambda_1 + \frac{\lambda_2}{b_2} \right) + \tilde{\pi} \left( \lambda_1, \frac{\lambda_2}{b_2} \right) \right).
$$
4.3. Transformation of the expressions for the limiting distributions to the Laplace domain

When we make the expression non-recursive we arrive at the expression for \( \pi(0,0) \), which is given by

\[
\pi(0,0) = \frac{\mu}{\lambda_1 + \lambda_2} \left[ \tilde{\pi} \left( \lambda_1 + \frac{\lambda_2}{b_2} \right) + \tilde{\pi} \left( \lambda_1, \frac{\lambda_2}{b_2} \right) \right].
\] (4.19)

As a result \( \tilde{\pi} \left( \lambda_1 + \frac{\lambda_2}{b_2} \right) \) denotes the Laplace transform \( \tilde{\pi}(s_1) \) evaluated at \( s_1 = \lambda_1 + \frac{\lambda_2}{b_2} \) and \( \tilde{\pi} \left( \lambda_1, \frac{\lambda_2}{b_2} \right) \) denotes the double Laplace transform \( \tilde{\pi}(s_1, s_2) \) evaluated at \( s_1 = \lambda_1 \) and \( s_2 = \frac{\lambda_2}{b_2} \).

### 4.3.2 Transformation of the expression for \( \pi(x, x) \)

We use the definition of the single Laplace transform to obtain the 1D distribution in the Laplace domain. We apply the definition on the left and right hand sides of Expression (4.16) and obtain

\[
\int_0^\infty \pi(x, x)e^{-s_1 x}dx = \frac{\mu(\lambda_1 + \frac{\lambda_2}{b_2})}{\mu + \lambda_1 + \lambda_2} \left( \int_0^\infty \pi(0,0)e^{-\frac{\mu + \lambda_1(1-b_2)}{b_2}x}e^{-s_1 x}dx + \int_0^\infty \int_0^\infty \pi(v, v)e^{-\frac{\mu + \lambda_1(1-b_2)}{b_2}(x-v)}dv e^{-s_1 x}dx + \int_0^\infty \int_0^\infty \pi(v, v)e^{-\lambda_1(1-b_2)}(\lambda_1 + \frac{\lambda_2}{b_2})(v-x)dv e^{-s_1 x}dx + \int_0^\infty \int_0^\infty \int_0^\infty \pi(v, w)e^{-\lambda_1(1-b_2)}(\lambda_1 + \frac{\lambda_2}{b_2})(v-w)dvdw e^{-s_1 x}dx + \int_0^\infty \int_0^\infty \int_0^\infty \pi(v, w)e^{-\lambda_1(1-b_2)}(\lambda_1 + \frac{\lambda_2}{b_2})(v-w)dvdw e^{-s_1 x}dx \right) (4.20)
\]

When we evaluate this expression we see that at the left hand side the definition of the single Laplace transform appears. The first term at the right hand side also integrates straight forward. Furthermore, we recognize a convolution in the second term. The third, fourth and fifth term are not that straight forward as the other terms. For a detailed derivation of these terms the reader is directed to Appendix C. When we work out the integrals in Expression (4.20) we obtain

\[
\tilde{\pi}(s_1) = \frac{\mu(\lambda_1 + \frac{\lambda_2}{b_2})}{\mu + \lambda_1 + \lambda_2} \left( \pi(0,0) \frac{1}{s + \frac{\lambda_1(1-b_2)}{b_2}} + \tilde{\pi}(s_1) \frac{1}{s + \frac{\lambda_1(1-b_2)}{b_2}} + \tilde{\pi} \left( \lambda_1 + \frac{\lambda_2}{b_2} \right) \frac{b_2}{b_2 s_1 - b_2 \lambda_1 - \lambda_2} - \tilde{\pi}(s_1) \frac{b_2}{b_2 s_1 - b_2 \lambda_1 - \lambda_2} + \tilde{\pi} \left( \lambda_1, s_1 - \lambda_1 \right) \frac{1}{s_1 + \frac{\lambda_1(1-b_2)}{b_2}} + \tilde{\pi} (\lambda_1, s_1 - \lambda_1) \left( \frac{1}{s_1 - (\lambda_1 + \frac{\lambda_2}{b_2})} - \tilde{\pi} (\lambda_1, \frac{\lambda_2}{b_2}) - \tilde{\pi} \left( \lambda_1, s_1 - \lambda_1 \right) \right) \right). (4.21)
\]
When we rewrite Expression (4.21) we obtain the expression for $\tilde{\pi}(s_1)$, which is given by

$$
\tilde{\pi}(s_1) = \frac{\mu B(s_1 - B) \left[ \pi(0, 0) + \tilde{\pi}\left(\lambda_1, s_1 - \lambda_1\right) \right] + \mu B(s_1 + A) \left[ \tilde{\pi}\left(\lambda_1 + \frac{\lambda_2}{\beta_2}\right) + \tilde{\pi}\left(\lambda_1, s_1 + \frac{\lambda_2}{\beta_2}\right) - \tilde{\pi}\left(\lambda_1, s_1 - \lambda_1\right) \right]}{(\mu + \lambda_1 + \frac{\lambda_2}{\beta_2})(s_1 + A)(s_1 - B) - \mu B(s_1 - B) + \mu B(s_1 + A)},
$$

(4.22)

with $A = \frac{\mu + \lambda_1(1 - \beta_2)}{\beta_2}$ and $B = \lambda_1 + \frac{\lambda_2}{\beta_2}$. We recognize the constants we already found in Expression (4.19), i.e. $\tilde{\pi}\left(\lambda_1 + \frac{\lambda_2}{\beta_2}\right)$ and $\tilde{\pi}\left(\lambda_1, s_1 - \lambda_1\right)$. Furthermore, we obtain $\tilde{\pi}\left(\lambda_1, s_1 - \lambda_1\right)$ which is the double Laplace transform $\tilde{\pi}(s_1, s_2)$ evaluated at $s_1 = \lambda_1$ and $s_2 = s_1 - \lambda_1$.

### 4.3.3 Transformation of the expression for $\pi(x, y)$

We use the definition of the double Laplace transform as defined in Expression (4.18) to obtain the 2D distribution in the Laplace domain. We apply the definition on the left and right hand sides of Expression (4.17) and obtain

$$
\int_0^\infty e^{-s_1 y} \int_y^\infty e^{-s_1 x} \pi(x, y) dxdy = \frac{\mu \lambda_1}{b_2} \int_0^\infty e^{-s_2 y} \int_y^\infty c^{-s_1 x} \pi(0, 0) e^{\lambda_1 x} e^{-\frac{s_1 + \lambda_1}{s_2} y} dxdy \\
+ \int_0^\infty e^{-s_2 y} \int_y^\infty c^{-s_1 x} \int_0^\infty \frac{e^{-s_1 x}}{\lambda_1 x} \pi(v, v) e^{-\lambda_1 (v - x)} e^{-\frac{s_1 + \lambda_1}{s_2} (y - v)} dvdxdy \\
+ \int_0^\infty e^{-s_2 y} \int_y^\infty c^{-s_1 x} \int_0^\infty \frac{e^{-s_1 x}}{\lambda_1 x} \pi(v, w) e^{-\lambda_1 (v - x)} e^{-\frac{s_1 + \lambda_1}{s_2} (y - w)} dvdwdxdy \\
+ \int_0^\infty e^{-s_2 y} \int_y^\infty c^{-s_1 x} \int_0^\infty \frac{e^{-s_1 x}}{\lambda_1 x} \pi(v, w) e^{-\lambda_1 (v - x)} e^{-\frac{s_1 + \lambda_1}{s_2} (y - w)} dvdwdxdy.
$$

(4.23)

When we evaluate this expression we observe that at the left hand side of Expression (4.23) the definition of the double Laplace transform appears. This is also in the first term of the right hand side of the expression. In the second, third and fourth term at the right hand side we obtain slightly complexer terms. We can work out the different terms and doing so we obtain

$$
\tilde{\pi}(s_1, s_2) = \frac{\mu \lambda_1}{b_2} \left( \frac{b_2(1 - b_2) \left( \pi(0, 0) + \tilde{\pi}(s_1 + s_2) + \tilde{\pi}(\lambda_1, s_1 + s_2 - \lambda_1) \right)}{(b_2(s_1 + s_2) + \mu + \lambda_1(1 - b_2))(s_1 + b_2 s_2 + \mu)} + \frac{b_2 \left( \tilde{\pi}(\lambda_1, s_1 + s_2 - \lambda_1) - \tilde{\pi}(s_1, s_2) \right)}{(s_1 - \lambda_1)(s_1 + b_2 s_2 + \mu)} \right).
$$

(4.24)

For a detailed derivation of the three terms in the right hand side of the expression the reader is directed to Appendix D. So, we arrive at the expression for $\tilde{\pi}(s_1, s_2)$ which is given by
4.4. Limiting distributions in the Laplace domain

\[ \tilde{\pi}(s_1, s_2) = \lambda \mu \left( \frac{(1 - 2b_2)s_1 - b_2s_2 - 2\lambda_1 + 2b_2\lambda_1 - \mu)\tilde{\pi}(s_1 + s_2)}{(s_1 - \lambda_1)(s_1 + b_2s_2 + \mu)(b_2(s_1 + s_2 - \lambda_1) + \lambda_1 + \mu)} + \frac{(1 - b_2)(s_1 - \lambda_1)\pi(0, 0) - (s_1 + b_2s_2 + \mu)\tilde{\pi}(\lambda_1, s_1 + s_2 - \lambda_1)}{(s_1 - \lambda_1)(s_1 + b_2s_2 + \mu)(b_2(s_1 + s_2 - \lambda_1) + \lambda_1 + \mu)} \right). \] (4.25)

We obtained this expression with help of Mathematica. As a result \( \tilde{\pi}(s_1 + s_2) \) denotes the Laplace transform \( \tilde{\pi}(s_1) \) evaluated at \( s_1 = s_1 + s_2 \) and \( \tilde{\pi}(\lambda_1, s_1 + s_2 - \lambda_1) \) denotes the double Laplace transform \( \tilde{\pi}(s_1, s_2) \) in which \( s_1 \) is fixed on \( \lambda_1 \) and \( s_2 = s_1 + s_2 - \lambda_1 \).

### 4.4 Limiting distributions in the Laplace domain

In the previous section we derived the expressions for the limiting distributions in the Laplace domain. The expressions that we need to solve to find the stationary waiting time distribution of the two class accumulating priority \( M/M/1 \) queue are

\[ \pi(0, 0) = \frac{\mu}{\lambda_1 + \lambda_2} \left[ \tilde{\pi}(\lambda_1 + \frac{\lambda_2}{b_2}) + \tilde{\pi}(\lambda_1, \frac{\lambda_2}{b_2}) \right], \] (4.26)

\[ \tilde{\pi}(s_1) = \frac{\mu B(s_1 - B) [\pi(0, 0) + \tilde{\pi}(\lambda_1, s_1 - \lambda_1)] + \mu B(s_1 + A) [\tilde{\pi}(\lambda_1 + \frac{\lambda_2}{b_2}) + \tilde{\pi}(\lambda_1, \frac{\lambda_2}{b_2}) - \tilde{\pi}(\lambda_1, s_1 - \lambda_1)]}{(\mu + \lambda_1 + \lambda_2)(s_1 + A)(s_1 - B) - \mu B(s_1 - B) + \mu B(s_1 + A)}, \] (4.27)

with \( A = \frac{\mu + \lambda_1(1 - b_2)}{b_2} \) and \( B = \lambda_1 + \frac{\lambda_2}{b_2} \),

and

\[ \tilde{\pi}(s_1, s_2) = \lambda_1 \mu \left( \frac{(1 - 2b_2)s_1 - b_2s_2 - 2\lambda_1 + 2b_2\lambda_1 - \mu)\tilde{\pi}(s_1 + s_2)}{(s_1 - \lambda_1)(s_1 + b_2s_2 + \mu)(b_2(s_1 + s_2 - \lambda_1) + \lambda_1 + \mu)} + \frac{(1 - b_2)(s_1 - \lambda_1)\pi(0, 0) - (s_1 + b_2s_2 + \mu)\tilde{\pi}(\lambda_1, s_1 + s_2 - \lambda_1)}{(s_1 - \lambda_1)(s_1 + b_2s_2 + \mu)(b_2(s_1 + s_2 - \lambda_1) + \lambda_1 + \mu)} \right). \] (4.28)

Some observations are made; we see that \( \pi(0, 0) \) is a point mass belonging to the idle period of the queue, \( \tilde{\pi}(s_1) \) is the 1D distribution belonging to the waiting time of the unaccredited customers and \( \tilde{\pi}(s_1, s_2) \) is the 2D distribution belonging to the waiting time of the accredited customers. From classical queueing theory we know that \( \pi(0, 0) = 1 - \rho \) and \( \rho = \frac{\lambda_1 + \lambda_2}{\mu} \). So, \( \tilde{\pi}(\lambda_1 + \frac{\lambda_2}{b_2}) + \tilde{\pi}(\lambda_1, \frac{\lambda_2}{b_2}) = \rho(1 - \rho) \). Furthermore, we find the roots for the denominator of Expression (4.27) to be \( s_1 = \lambda_1 \) and \( s_1 = -(\lambda_1 + \mu) \frac{1 - 2b_2}{b_2} \), in the special case that \( \lambda_2 = b_2\mu \) and \( \lambda_1 < (1 - b_2)\mu \). So, when we substitute \( s_1 = \lambda_1 \) into the numerator the result should be zero since the Laplace transform needs to be analytic within the right half plane. However, we cannot obtain the expression for \( \tilde{\pi}(\lambda_1, s_1 - \lambda_1) \) due to the fact that when we substitute \( s_1 = \lambda_1 \) we obtain the constant \( \tilde{\pi}(\lambda_1, 0) \). We can obtain the expression for this constant but not for the
unknown function.

We cannot obtain the unknown functions in the expressions, therefore we try to obtain the stationary waiting time distribution via another way. With the information above we have two expressions left to solve. Since, the unknown functions in the expressions are the known expressions evaluated on different points we know the expressions we need to solve.

The constants $A$, $B$, $\pi(0,0)$ and $\tilde{\pi}(\lambda_1 + \frac{\lambda_2}{s_2}) + \tilde{\pi}(\lambda_1, \frac{\lambda_2}{s_2})$ are substituted in Expression (4.27) and (4.28) and simplified with Mathematica. The expressions that are obtained and should be solved are:

\[
\tilde{\pi}(s_1) = \frac{(b_2 \lambda_1 + \lambda_2)((b_2(s_1 - \lambda_1) + \lambda_1)(\lambda_1 + \lambda_2 - \mu) + \mu^2(\tilde{\pi}(s_1) + \tilde{\pi}(\lambda_1, s_1 - \lambda_1)))}{(b_2(s_1 - \lambda_1) - \lambda_2)\mu(b_2(s_1 - \lambda_1) + \lambda_1 + \mu)}
\] (4.29)

and

\[
\tilde{\pi}(s_1, s_2) = \lambda_1 \mu \left( \frac{((1 - 2b_2)s_1 - b_2s_2 - 2\lambda_1 + 2b_2\lambda_1 - \mu)\tilde{\pi}(s_1 + s_2)}{(s_1 - \lambda_1)(s_1 + b_2s_2 + \mu)(b_2(s_1 + s_2 - \lambda_1) + \lambda_1 + \mu)} + \frac{(1 - b_2)(s_1 - \lambda_1)\frac{\lambda_1 + \lambda_2}{\mu} - (s_1 + b_2s_2 + \mu)\tilde{\pi}(\lambda_1, s_1 + s_2 - \lambda_1)}{(s_1 - \lambda_1)(s_1 + b_2s_2 + \mu)(b_2(s_1 + s_2 - \lambda_1) + \lambda_1 + \mu)} \right).
\] (4.30)

To solve these expressions the following approach is proposed:

1. We take Expression (4.30). This expression has two unknown expressions, i.e. $\tilde{\pi}(s_1 + s_2)$ and $\tilde{\pi}(\lambda_1, s_1 + s_2 - \lambda_1)$. Substitution of $s_1 = \lambda_1$ and $s_2 = s_1 - \lambda_1$ should be applied. From observation of the unknown expressions in Expression (4.30) we know that we should obtain an recursive expression for $\tilde{\pi}(\lambda_1, s_1 - \lambda_1)$ that has the unknown expression $\tilde{\pi}(s_1)$ in the right hand side.

2. This expression should be made non recursive. Now, an expression for $\tilde{\pi}(\lambda_1, s_1 - \lambda_1)$, that depends on the unknown expression $\tilde{\pi}(s_1)$, is obtained.

3. The expression for $\tilde{\pi}(\lambda_1, s_1 - \lambda_1)$ should be substituted in Expression (4.29). A recursive expression for $\tilde{\pi}(s_1)$ is obtained.

4. By making this expression non recursive we should obtain the stationary waiting time distribution for the unaccredited customers in the two class accumulating priority queue in the Laplace domain.

5. A kind of similar approach could be applied to obtain the stationary waiting time distribution for the accredited customers in the Laplace domain.

However, when we use this approach and Expression (4.30) is made non recursive and when we substitute $s_1 = \lambda_1$ and $s_2 = s_1 - \lambda_1$ we obtain the identity
4.4. Limiting distributions in the Laplace domain

\[ \tilde{\pi}(\lambda_1, s_1 - \lambda_1) = \tilde{\pi}(\lambda_1, s_1 - \lambda_1). \] (4.31)

So, the proposed approach of solving the expressions cannot be applied. Also, by taking the expression for \( \tilde{\pi}(s_1, s_2) \) as the starting point of the approach we are not able to solve the expressions. So, up to this moment we are unable to solve the two expressions and obtain the stationary waiting time distributions of the accredited and unaccredited customers in the two class accumulating priority queue analytically. More research should be conducted and probably other ways to derive the stationary waiting time distribution of the two class accumulating priority queue should be applied or derived.
Chapter 4. Study of the two class maximum priority process in the $M/M/1$ queue
Chapter 5

Numerical tools

Until now, we are not able to obtain the stationary waiting time distributions in the two class accumulating priority queue analytically. But we are able to obtain the distribution numerically by simulation. In this chapter we present two numerical tools that are of interest in the study of the accumulating priority queue. In order to observe how an \(M/M/1\) queue with different customer classes which accumulates priority with different rates reacts in practice, we have written an m-file in MATLAB containing a simulation program. In the first section of this chapter we explain our simulation procedure.

In our research towards the stationary distribution of the accumulating priority queue Laplace transforms are used. In order to compare the theoretical waiting time distribution of the accredited and unaccredited customers with the waiting times we obtain by simulation, we need to make a conversion from the Laplace domain to the time domain. It is possible that complex Laplace transforms are obtained. Therefore a numerical inversion of Laplace transforms m-file is written. We explain this numerical inversion in the second section of this chapter.

5.1 Simulation procedure

When we write a simulation of the accumulating priority queue we could choose between two options. The first option is to apply a standard event-scheduling approach, when we apply such an approach we have to maintain a record of all customers in the queue together with their accumulated priorities during the whole time-span over which we simulate. The computational costs are high in comparison with the alternative simulation method which simulates only the maximum priority process. This method requires only that we record the maximal priorities for each of the customer classes, the length of the current service time and the time that it commenced [1]. The computational costs for the latter method are relatively lower than for the first method. Therefore, we choose the method in which we simulate the maximum priority process.

The reader will find an explanation of the simulation method in this section. First, we will discuss the different declarations we make in the beginning of the simulation. After the declarations the simulation procedure is given and in the end of the section we provide the simulation
output. The complete m-file for the simulation procedure can be found in Appendix E.

5.1.1 Declarations

The simulation m-file we wrote simulates the maximum priority process in the two class accumulating priority queue, this simulation file can easily be extended to more classes if necessary. As before, we consider an $M/M/1$ queue in which the service time distributions of classes 1 and 2 are the same. We start our simulation program with the declaration of the number of customer classes, arrival rates and priority rates of the different classes, the total number of customers entering the system and the mean of the service time distribution, which is the same for both classes. Furthermore, we make use of the variables $\tau(m)$, $T_{\text{Arrival}}(m)$, $X(m)$ and $D(m)$ which denote the inter-arrival, arrival, service and departure time of the $m^{th}$ customer, respectively. Besides these (known) variables, we introduce a new variable $E(m,k)$ in this variable we store a randomly drawn number which we use to determine whether the next customer to be served is an accredited, unaccredited or no customer at all. The $k$ in $E(m,k)$ denotes the customer class, so in the simulation $k = 1, 2$. It is not necessary to keep track of all these variables during the simulation, but for analysis purposes we made the choice to keep track of them. We introduce a boolean called $b\text{BusyPeriod}$, with this boolean we indicate whether the simulation is in a busy period or not.

In the simulation code we make use of an array $\text{Time}$ and a matrix $M$, we do not know the length of this array and matrix. It depends on the number of busy periods that occur during the simulation. So, we cannot pre-allocate any memory since we do not know the number of these periods on beforehand. We use the array $\text{Time}$ to store the time points that are of importance for the simulation, i.e. the time points of arrival of a customer at the server. If the system is in a busy period, the arrival time point of the $(m + 1)^{th}$ customer is equal to the time point at which the $m^{th}$ customer departs from the server. Note that we store the time points in duplicate in the array. The reason lies in the matrix $M$ in which we store the maximal priorities at the specific time points for the different customer classes. When a customer leaves the server a drop in the maximal priority is observed. This drop occurs when a customer departs from the server and happens at the specific time point. To keep track of this drop we store the maximal priority at the time points $D(m)−$ (the time instant just before the drop) and $D(m)+$ (the time instant just after the drop) in the matrix $M$. Furthermore, we use a matrix instead of an array to store the accumulated priorities because of the different customer classes. In the two class accumulating priority queue we use the first row of $M$ to store the accumulated priorities belonging to class 1 customers and the second row for the class 2 customers.

5.1.2 Simulation procedure

This section contains the pseudo code of the simulation. We use the following simulation procedure

\begin{verbatim}
for m = 1 to total number of customers do
    if boolean $b\text{BusyPeriod}$ false then

\end{verbatim}
Draw $\tau(m)$ from exponential distribution with mean $(\lambda_1 + \lambda_2)^{-1}$.

5: Determine arrival time, $T_{\text{Arrival}}(m)$ is equal to the latest recorded time point plus $\tau(m)$.
Store arrival time in $\text{Time}$ in duplicate.
Store priorities equal to zero in matrix $M$.

Draw $X(m)$ from exponential distribution with mean $\mu^{-1}$.

10: Determine departure time, $D(m)^+ = T_{\text{Arrival}}(m) + X(m)$.
Store $D(m)^+$ in $\text{Time}$ in duplicate.
Determine priorities at $D(m)^+$, priorities equal to $b_i X(m)$ with $i = 1, 2$.
Store maximal priority at $D(m)^+$ in $M$.

else

15: Determine arrival time, $T_{\text{Arrival}}(m) = D(m - 1)$.
Service time is known, see next if statement.
Determine departure time, $D(m)^+ = T_{\text{Arrival}}(m) + X(m)$.
Store $D(m)^+$ in $\text{Time}$ in duplicate.
Determine priorities at $D(m)^-$, priorities are equal to $b_i X(m)$ plus the latest recorded priority of class $i = 1, 2$.

end if

Draw $E(m, 1)$ from exponential distribution with mean $\frac{\lambda_1}{\mu}$.

25: if $E(m, 1) < M(1, \text{length}(M)) - M(2, \text{length}(M))$ then

Boolean $b_{\text{BusyPeriod}}$ (becomes) true.
Next customer is an accredited customer.
Determine priorities at $D(m)^+$, priority of class 1 is equal to the latest recorded priority of class 1 minus $E_{m, 1}$ and the priority of class 2 does not change, so it is equal to the latest recorded priority of class 2.

30: Draw $X(m + 1)$ from exponential distribution with mean $\mu^{-1}$.
else

Draw $E(m, 2)$ from exponential distribution with mean $(\frac{\lambda_1}{b_1} + \frac{\lambda_2}{b_2})^{-1}$.

35: if $E(m, 2) < M(2, \text{length}(M))$ then

Boolean $b_{\text{BusyPeriod}}$ (becomes) true.
Next customer is an unaccredited customer.
Determine priorities at $D(m)^+$, priorities of both classes are equal to the latest recorded priority of the class 2 customer minus $E(m, 2)$.

40: Draw $X(m + 1)$ from exponential distribution with mean $\mu^{-1}$.
else

Boolean $b_{\text{BusyPeriod}}$ (becomes) false.
The priorities at $D(m)^+$ are set to zero for both classes.

45: end if

end if

end for
5.1.3 Output

To get insight in the accumulating priority queue in the simulation, we plot the maximum priority process against time. We use the matrix $M$ and array $Time$ for this purpose, an illustration is depicted in Figure 5.1, we obtained this figure after a simulation with 20 customers and two different customer classes, i.e. class one and two. The simulation parameters that are used are $\lambda_1 = 1, \lambda_2 = 2, \mu = 4, b_1 = 1$ and $b_2 = 0.5$. We see unaccredited customers starting their service around the time points 1.6, 3.5, 3.7, 6.4, 8.0, 8.4 and 9.6. Furthermore, in the last busy period in the figure, we see accredited customers going into service around the time points 7.1, 7.2, 7.9 (two customers) and 9.3. In total we can distinguish 8 busy periods (there is a very small busy period around time point 5.3).

![Maximal priorities versus time](image)

Figure 5.1: Sample path of the maximum priority process, obtained with a simulation with two different customer classes; simulation parameters $\lambda_1 = 1, \lambda_2 = 2, \mu = 4, b_1 = 1$ and $b_2 = 0.5$.

Furthermore, we are interested in the accumulated priorities at the time points the customers commence service. We find these accumulated priorities in the even entries of the matrix $M (D(m)^+)$. When we plot the accumulated priorities of the class two customers against the accumulated priorities of the class one customers for the simulation with 20 customers we obtain the $(M_2(t), M_1(t))$ plot as depicted in Figure 5.2a. From analysis of the maximum priority process we know that the point mass in the origin describe the probability of an idle period $(1 - \rho)$. On the line $M_2(t) = M_1(t)$ we find the accumulated priorities belonging to unaccredited customers entering service. The accumulated priorities we allocate in the wedge between the lines $M_2(t) = M_1(t)$ and $M_2(t) = b_2 M_1(t)$ belong to the accredited customers who commence service. In Figures 5.2b and 5.2c $(M_2(t), M_1(t))$ plots can be found for respectively simulations of 100 and 10000 customers.

![Maximal priorities versus time](image)

Figure 5.2a: $(M_2(t), M_1(t))$ plot for a simulation with 20 customers.

We are able to find the waiting times of the unaccredited and accredited customers if we take the accumulated priorities at the moment the customer enters service into consideration. Because,
5.1. Simulation Procedure

if we have an accredited customer entering service than the accumulated class one priority will be strictly greater then the accumulated class two priority. We know that an unaccredited customer is going into service when both the class one and two accumulated priorities are the same. So, we can make a distinction between the accredited and unaccredited customers by observation of the matrix $M$ at the even entries of this matrix. To obtain the waiting times we go back to Section 2.4, in that section we pointed out some observations about the accumulated priority process and especially about accredited and unaccredited customers. We know that accredited customers are only class one customers which accumulate priority with rate $b_1$. So, the waiting time of an accredited customer is equal to

$$W_{\text{Accredited}} = \frac{M_1}{b_1}.$$  \hspace{1cm} (5.1)

For the unaccredited customers we know that these are a mixture of class one and two customers, both gaining priority with a different priority rate. So, to obtain the waiting time of an unaccredited customer we need to sample the class of the customer and then dividing his accumulated priority by his priority rates, i.e. $b_1$ or $b_2$. 

Figure 5.2: The $(M_2, M_1)$ plots for different amount of customers.
We wrote an m-file to obtain the waiting times of the accredited and unaccredited customers. This m-file is straightforward and can be found in Appendix F. With the obtained waiting times we can construct the probability density of the waiting times for the different customer types, i.e. accredited or unaccredited. In Figures 5.3a and 5.3b the probability densities are plotted for the accredited and unaccredited customers respectively. We create these probability densities by making a histogram of the waiting time data we obtained from the simulation. Since the area of a probability density must be equal to one, we divide the obtained histogram by its area that we obtain by integration via the trapezoidal rule. As shown in the plots the number of bins used to create the histograms is different for the accredited and unaccredited customers. We can also see that the probability density for the waiting times of the unaccredited customers is smoother than the probability density of the accredited customer. The reason for this is the number of accredited and unaccredited customers that enter the system during simulation. In total 250000 customers enter the system; 31183 accredited, 156861 unaccredited and 61956 customers that initiate a busy period. Due to the fact that the amount of accredited customers is lower than unaccredited customers we use lesser bins to create the probability density of the accredited customers. The m-file we use to create the probability densities from Figures 5.3a and 5.3b can be found in Appendix G.

Figure 5.3: These plots show the probability densities of the waiting times for the accredited and unaccredited customers obtained from a simulation where 10000 customers enter the two class accumulating priority queue.

(a) PDF for waiting times accredited customers, histogram contains 50 bins.  
(b) PDF for waiting times accredited customers, histogram contains 100 bins.

5.2 Numerical inversion of Laplace transforms

We want to compare the theoretical waiting time distribution of the accredited and unaccredited customers with the waiting time distribution we obtain with the m-files we wrote to simulate the maximum priority process in the two class accumulating priority M/M/1 queue, as presented in the previous section. In order to make a comparison we need to make a conversion of the theoretical distribution from the Laplace domain to the time domain. Assuming that we can solve the equations in the Laplace Domain and we cannot invert the expressions analytically.
Especially, when further research is done and the method we developed is applied on more complex queue types, for example $M/G/1$ queues, the numerical inversion algorithms could be of use. In this section we present two different algorithms for the numerical inversion designed by Abate and Whitt [3]. It is beyond the scope of this report to show the whole derivation of the expressions used in the algorithms. Instead we give the expressions for the two algorithms and the reader is directed to the paper of Abate and Whitt for the derivation.

Abate and Whitt worked out the two algorithms especially for probability cumulative distribution functions, but they apply also to other functions. The reason two different algorithms are proposed is because it does not seem possible to provide effective methods with simple general error bounds that are independent of the function we take into consideration. The two different algorithms that are proposed are assumed to agree both within the desired precision [3].

The two methods are both variants of the Fourier-series method, but very different. Thus, we expect that the algorithms serve as useful checks on each other. The Fourier-series method can be interpreted as numerically integrating a standard inversion integral by means of the trapezoidal rule. The key mathematical result is the Poisson summation formula, which identifies the discretization error associated with the trapezoidal rule and thus helps bound it.

### 5.2.1 The Euler method

The first method provided by Abate and Whitt [3] is called the Euler method since we employ an Euler summation, based on the Bromwich contour inversion integral. This integral can be expressed as the integral of a real-valued function of a real variable by choosing a specific contour. The Euler summation is given by

\[
E(m, n, t) = \sum_{k=0}^{m} \binom{m}{k} 2^{-m} s_{n+k}(t). \tag{5.2}
\]

Thus, $E(m, n, t)$ is the binomial average of the terms $s_n, s_{n+1}, \ldots, s_{n+m}$. Typically values of $m = 11$ and $n = 15$ are chosen, increasing $n$ if necessary. $s_n(t)$ is the approximation of the function we want to numerically invert, this approximation is given by

\[
s_n(t) = \frac{e^{A/2}}{2t} \text{Re}(\hat{f}) \left(\frac{A}{2t}\right) + \frac{e^{A/2}}{t} \sum_{k=1}^{n} (-1)^k a_k(t), \tag{5.3}
\]

where

\[
a_k(t) = \text{Re}(\hat{f}) \left(\frac{A + 2k\pi i}{2t}\right). \tag{5.4}
\]

In Expressions (5.3) and (5.4), $\hat{f}$ denotes the Laplace function we want to invert. If we want to have at most $10^{-\gamma}$ discretization error, we let $A = \gamma \log 10$. Abate and Whitt often use $A = 18.4$ to achieve a $10^{-8}$ discretization error. In order to estimate the error associated with
Chapter 5. Numerical tools

Euler summation Abate and Whitt [3] suggest to use the difference between successive terms, i.e. \( E(m, n + 1, t) - E(m, n, t) \), which is usually a good error estimate.

Abate and Whitt provide an implementation of the algorithm based on the Euler method in their paper, they wrote the algorithm in UBASIC. We write the algorithm in MATLAB and our implementation is provided in Appendix H.

5.2.2 The Post-Widder method

The second method Abate and Whitt describe in their paper is a method based on the Post-Widder Theorem, which expresses the function we want to obtain in the time domain \((f(t))\) as the pointwise limit as \( n \to \infty \) of

\[
f_n(t) = \frac{(-1)^n}{n!} \left( \frac{n + 1}{t} \right)^{n+1} \hat{f}(n) \left( \frac{n + 1}{t} \right),
\]

where \( \hat{f}(n)(s) \) denotes the \( n^{th} \) derivative of the Laplace transform \( \hat{f} \) at \( s \).

For the Euler summation based on the Post-Widder theorem Abate and Whitt use a linear combination, i.e.

\[
\tilde{f}_{j,m}(t) = \sum_{k=1}^{m} w(k, m) f_{jk}(t),
\]

starting at \( j = 10 \) and \( m = 6 \) and increasing when necessary. The general weights \( w(k, m) \) are given by

\[
w(k, m) = (-1)^{m-k} \frac{k^m}{k!(m-k)!}.
\]

\( f_{jk}(t) \) in Expression (5.6) is the Post-Widder theorem given in Expression (5.5) where discrete Poisson summation, to obtain the trapezoidal rule approximation, is applied (with step size \( \pi/n \) and the associated error bound) is written as

\[
f_n(t) = \frac{n + 1}{2knm} \left\{ \hat{f} \left( \frac{(n + 1)(1 - r)}{t} \right) + (-1)^n \hat{f} \left( \frac{(n + 1)(1 + r)}{t} \right) + 2 \sum_{k=1}^{n-1} (-1)^k Re(\hat{f}) \left( \frac{n + 1}{t} (1 - re^{\pi tk/n}) \right) \right\} - e_d,
\]

where

\[
e_d = \sum_{j=1}^{\infty} f_{n+jm} \left( t + \frac{tj2m}{n+1} \right)^2 j^m.
\]
In Expressions (5.8) and (5.10) \( r \) denotes the radius of the Cauchy contour integral. This integral is used as the inversion integral in this algorithm. Assuming \( |f(t)| \leq 1 \) for all \( t \), we have \( |f_n(t)| \leq 1 \) for all \( n \) and \( t \), so that [3]

\[
|e_d| \leq \frac{r^{2n}}{1 - r^{2n}} \approx r^{2n}.
\]

As with the algorithm based on the Euler method, Abate and Whitt provide an implementation of the Post-Widder algorithm in their paper written in UBASIC. We write the algorithm in MATLAB and our implementation is provided in Appendix I.
Chapter 6

Conclusion and recommendations

In this chapter the conclusions and recommendation of the research are provided. In the first section the reader will find the conclusion while some recommendations for further research are given in the second section.

6.1 Conclusion

The goal of this project was to derive the stationary waiting time distribution of the accumulating priority queue. Therefore, we developed a framework to derive the stationary waiting time distribution of the maximum priority process in the two class accumulating priority $M/M/1$ queue. We assumed the customers arriving according to a Poisson process with $\lambda_1$ and $\lambda_2$, the arrival rates of the first and second customer class, respectively. Furthermore, the assumption is made that both customer classes received the same exponential distributed service ($\mu$). The priority rates of the customers where set on $b_1$ (for class 1 customers) and $b_2$ (for class 2 customers) with $b_2 < b_1$ and $b_1 = 1$.

The accumulating priority queue is coupled to the maximum priority process, which is a continuous-time process. To derive the stationary waiting time distribution analytically a framework is developed. This framework is developed with the one class accumulating priority $M/M/1$ queue and the maximum priority process. The stationary waiting time distribution of this queue is known. Because this distribution is equal to the stationary waiting time distribution of the ordinary $M/M/1$ queue. The framework that is developed consists of the following steps:

1. Derive the discrete time process ($M^n$) (or $(M^n_1,M^n_2)$ in the two class case) by looking at the embedded points of the (continuous-time) maximum priority process where customers enter service. This discrete time process can be visualized in a Markov model. From this model the transition probabilities, probability densities and probability density functions are derived.

2. Define the expressions for the limiting distributions of the Markov model. In these expressions unknown expressions (which are the unknown waiting time distributions) will appear.
3. Use the Laplace transform to write the expressions for the limiting distributions in the Laplace domain.

4. Solve the expressions for the limiting distributions in the Laplace domain. The stationary waiting time distribution for a specific class of the accumulating priority queue should now be obtained.

The framework works perfectly for the one class accumulating priority queue. We are able to derive the transition probabilities and densities of the Markov model and we can set up the expressions for the limiting distributions. When these expressions are transformed to the Laplace domain a constant (which is the waiting time distribution evaluated at a certain point) appears in the numerator of the expression for the waiting time distribution in the Laplace domain. By evaluation of the denominator a root that is equal to 0 is observed. Since a Laplace transform has to be analytic for $\text{Re}(s) \geq 0$ this root need to appear in the numerator as well. With this knowledge the constant is derived. Now, the stationary waiting time distribution of the one class accumulating priority is obtained. The method we derived is also another way to derive the stationary waiting time distribution of the $M/M/1$ queue. In classical queueing theory, this distribution is obtained by analysis of the virtual workload process. We proved that deriving this distribution can also be done by analyzing the maximum priority process.

To obtain the stationary waiting time distribution for the two class accumulating priority $M/M/1$ queue the same framework as for the one class queue is used. The difference is that instead of the discrete time process $(M^n)$ the discrete time process $(M^n_1, M^n_2)$ is derived. Furthermore, the stationary waiting time distribution of the unaccredited customers will be a 1D distribution while a 2D distribution is derived for the accredited customers. The framework works perfectly until we try to solve the expressions for the limiting distributions in the Laplace domain. We try to solve these expressions by substitution. However, when we apply this technique an identity is obtained. We cannot go any further in our derivation than this identity. So, until now we are able to derive the expressions for the limiting distributions of the two class accumulating priority queue. However, we are not able to actually obtain the stationary waiting time distribution of this queue with our framework.

We are not able to obtain the stationary waiting time distribution of the two class accumulating priority queue analytically. To get some insight in the stationary waiting time distributions of the accredited and unaccredited customers, a simulation procedure is written in Matlab. With this simulation we are able to plot the maximum priority process in the accumulating priority queue and obtain the distributions numerically. Furthermore, numerical inversion m-files are written in Matlab for the numerical inversion of complex Laplace transforms. These files could be of assistance when complex distributions are obtained in a further stage of the research.

6.2 Recommendations

Until now, we did not succeed in obtaining the stationary waiting time distribution of the two class accumulating priority queue with the framework that is proposed in this report. We were able to obtain the expressions for the limiting distributions which should give the stationary waiting time distribution, once solved. However, it is not possible to solve these expressions.
Therefore, more effort should be put into deriving other methods to obtain the stationary waiting time distribution.

In general, it should be possible to derive the stationary waiting time distributions for the accumulating priority $M/M/1$ queue since the arrivals and services are exponential distributed. Apparently, the $2D$ distribution puts some extra complexity in the expressions that we cannot solve with our approach. Further investigation could also be done to the $2D$ distribution. Problems where $1D$ distributions arise in the $M/M/1$ queue are possible to solve (see Chapter 3). At the moment $2D$ distributions arise the complexity of the problem grows rapidly.

If the problems that we encounter can be solved and a method to derive the stationary waiting time distribution in the two class accumulating priority $M/M/1$ queue is obtained this method should be applied on other queues. It is of interest to investigate the stationary waiting time distribution of the $M/G/1$ queue with multiple classes. Since, this queue type is more realistic than the $M/M/1$ queue. It is also possible to investigate other priority strategies. For instance, when customers enter service their priority accumulates from a certain starting priority which is greater than 0. This is, however, still just the tip of the iceberg. It should be clear that the accumulating priority queue is a very interesting queue. Still a lot of research can be conducted on this queue type.
List of Figures

2.1 The accumulating priority model with two customer classes [1]. . . . . . . . . . . . 4
2.2 The virtual workload and maximum waiting time processes for an queueing sys-
    tem. The virtual workload process is depicted in the upper and the maximum 
    waiting time process in the lower plot [1]. . . . . . . . . . . . . . . . . . . . . . . 7
2.3 The maximum priority process in the two class accumulating priority queue. . . . 9
3.1 The Markov Chain of the maximum priority process with one customer class. . . 12
3.2 These plots show the maximum priority process when the different transitions in 
    the MC occurs. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 14
4.1 The Markov Chain of the maximum priority process with two different customer 
    classes. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 22
4.2 These plots show the maximum priority process when the different transitions 
    from the idle state in the MC occurs. . . . . . . . . . . . . . . . . . . . . . . . . . 24
4.3 These plots show the maximum priority process when the different transitions 
    from the accredited service set of states in the MC occurs. . . . . . . . . . . . . 27
4.4 These plots show the maximum priority process when the different transitions 
    from the unaccredited service set of states in the MC occurs. . . . . . . . . . . . 29
4.5 Plot of $M_2(t)$ versus $M_1(t)$ in 4.5a and with the transition bounds for the third 
    expression in 4.5b, transitions to the point $(x,y)$ are only possible form the origin 
    (idle), red line (unaccredited) or areas I and II (accredited). . . . . . . . . . . . . 31
5.1 Sample path of the maximum priority process, obtained with a simulation with 
    two different customer classes; simulation parameters $\lambda_1 = 1, \lambda_2 = 2, \mu = 4, b_1 = 1$ 
    and $b_2 = 0.5$. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 42
5.2 The $(M_2, M_1)$ plots for different amount of customers. . . . . . . . . . . . . . 43
5.3 These plots show the probability densities of the waiting times for the accredited 
    and unaccredited customers obtained from a simulation where 10000 customers 
    enter the two class accumulating priority queue. . . . . . . . . . . . . . . . . . . . 44
Bibliography


Appendix A

Proof of the convolution theorem

The convolution theorem is given by the following expression

\[ F_1(s)F_2(s) = \mathcal{L}\{f_1(t) \ast f_2(t)\} = \int_0^\infty e^{-st} \int_0^t f_1(t-\tau)f_2(\tau)d\tau dt. \quad (A.1) \]

The right hand side of this expression is given by

\[ \int_0^\infty e^{-st} \left( \int_0^t f_1(\tau)f_2(t-\tau)d\tau \right) dt, \]

which, by changing the order of integration, becomes

\[ \int_0^\infty f_1(\tau) \left( \int_\tau^\infty f_2(t-\tau)e^{-st}dt \right) d\tau. \]

We can write this as

\[ \int_0^\infty f_1(\tau) \left( \int_0^\infty f_2(t)e^{-s(t+\tau)}dt \right) d\tau \]
\[ = \int_0^\infty f_1(\tau)e^{-st}d\tau \int_0^\infty f_2(t)e^{-st}dt \]
\[ = F_1(s)F_2(s). \]

So, Expression (A.1) holds.
Appendix A. Proof of the convolution theorem
Appendix B

Check for transition probability and density functions in the two class problem

We start with verifying Expression (4.12). When we substitute the relevant probability and density functions in the left hand side of this expression we obtain

\[
p_{0,0 \rightarrow 0,0} + \int_0^\infty \int_y^{\infty} f_{0,0 \rightarrow x,y} dx dy + \int_0^\infty f_{0,0 \rightarrow x,x} dx
\]

\[
= \frac{\mu}{\mu + \lambda_1 + \lambda_2} + \int_0^\infty \int_y^{\infty} \frac{\mu \lambda_1}{b_2} e^{\lambda_1 x} e^{-(\mu + \lambda_1) \frac{x^2}{2}} dx dy + \int_0^\infty e^{-(\mu + \lambda_1 (1-b_2)) x} \frac{\mu (\lambda_1 + \frac{\lambda_2}{b_2})}{\mu + \lambda_1 + \lambda_2} dx
\]

\[
= \frac{\mu}{\mu + \lambda_1 + \lambda_2} + \frac{\mu}{b_2} \int_0^\infty (e^{\lambda_1 y} - e^{\lambda_2 y}) e^{-(\mu + \lambda_1) \frac{y^2}{2}} dy + \frac{\mu (\lambda_1 + \frac{\lambda_2}{b_2})}{\mu + \lambda_1 + \lambda_2} \frac{b_2}{\mu + \lambda_1 + \lambda_2 (\mu + \lambda_1 (1-b_2))}
\]

\[
= \frac{\mu}{\mu + \lambda_1 + \lambda_2} + \frac{1}{\mu + \lambda_1 (1-b_2)} + \frac{\mu (\lambda_1 + \frac{\lambda_2}{b_2})}{(\mu + \lambda_1 (1-b_2)) (\mu + \lambda_1 + \lambda_2)}
\]

\[
= 1 + \frac{\mu (\mu + \lambda_1 (1-b_2)) + \mu (\lambda_1 + \frac{\lambda_2}{b_2})}{(\mu + \lambda_1 (1-b_2)) (\mu + \lambda_1 + \lambda_2)} - \frac{\mu (\mu + \lambda_1 + \lambda_2)}{(\mu + \lambda_1 (1-b_2)) (\mu + \lambda_1 + \lambda_2)}
\]

\[
= 1 + \frac{(\mu + \lambda_1 (1-b_2)) (\mu + \lambda_1 + \lambda_2)}{(\mu + \lambda_1 (1-b_2)) (\mu + \lambda_1 + \lambda_2)} - \frac{\mu (\mu + \lambda_1 + \lambda_2)}{(\mu + \lambda_1 (1-b_2)) (\mu + \lambda_1 + \lambda_2)}
\]

\[
= 1 - \frac{\mu (\mu + \lambda_1 + \lambda_2)}{(\mu + \lambda_1 (1-b_2)) (\mu + \lambda_1 + \lambda_2)}
\]

So, the transition probability and densities from the Idle Period state integrate as they should. The next equality to be verified is Expression (4.13). When we substitute the relevant probability and density functions in the left hand side of this expression we obtain

59
\[ p_{v,w=0,0} + \int_{w}^{\infty} \int_{y}^{x} f_{v,w \to x,y} \, dy \, dx + \int_{0}^{w} f_{v,w \to x,x|x \leq w} \, dx + \int_{w}^{\infty} f_{v,w \to x,x|x \geq w} \, dx \]

\[ = e^{-\lambda_1 v} \frac{\lambda_2 w}{\mu + \lambda_1 + \lambda_2} + \int_{w}^{\infty} \int_{y}^{x} \frac{\mu \lambda_1}{b_2} e^{-\lambda_1 (v-x)} e^{-(\mu + \lambda_1) \frac{y-w}{b_2}} \, dy \, dx \]

\[ + \int_{0}^{w} e^{-\lambda_1 (v-w)} e^{-(\lambda_1 + \frac{\lambda_2}{b_2}) (w-x)} \frac{\mu (\lambda_1 + \frac{\lambda_2}{b_2})}{\mu + \lambda_1 + \lambda_2} \, dx + \int_{w}^{\infty} e^{-\lambda_1 (v-w)} e^{-\frac{\mu + \lambda_1 (1-b_2)}{b_2} (x-w)} \frac{\mu (\lambda_1 + \frac{\lambda_2}{b_2})}{\mu + \lambda_1 + \lambda_2} \, dx \]

\[ = e^{-\lambda_1 v} \frac{\lambda_2 w}{\mu + \lambda_1 + \lambda_2} + \int_{w}^{\infty} \frac{\mu}{b_2} e^{-\frac{\mu - w}{b_2} e^{-\lambda_1 (v-w)} e^{-(\lambda_1 + \frac{\lambda_2}{b_2}) w} + e^{-\lambda_1 (v-w)} \frac{\mu (\lambda_2 + \lambda_1 b_2)}{(\mu + \lambda_1 + \lambda_2) (\mu + \lambda_1 (1-b_2))} + e^{-\lambda_1 (v-w)} \frac{\mu \lambda_2 + \lambda_1 b_2}{\mu + \lambda_1 + \lambda_2}} \frac{\mu \lambda_1 + \lambda_1 b_2}{\mu + \lambda_1 + \lambda_2} \frac{\mu (\lambda_1 + \lambda_1 b_2)}{(\mu + \lambda_1 + \lambda_2) (\mu + \lambda_1 (1-b_2))} \]

\[ = 1 - \frac{\mu \lambda_1 + \lambda_1 b_2}{(\mu + \lambda_1 + \lambda_2) (\mu + \lambda_1 (1-b_2))} e^{-\lambda_1 (v-w)} + \frac{\mu}{\mu + \lambda_1 + \lambda_2} e^{-\lambda_1 (v-w)} + e^{-\lambda_1 (v-w)} \frac{\mu (\lambda_2 + \lambda_1 b_2)}{(\mu + \lambda_1 + \lambda_2) (\mu + \lambda_1 (1-b_2))} \]

\[ = 1 + \frac{\mu \lambda_1 + \lambda_1 b_2}{(\mu + \lambda_1 + \lambda_2) (\mu + \lambda_1 (1-b_2))} + \frac{\mu}{\mu + \lambda_1 + \lambda_2} - \frac{\mu}{\mu + \lambda_1 + \lambda_2} e^{-\lambda_1 (v-w)} \]

\[ = 1 + \left( \frac{\mu \lambda_1 + \lambda_1 b_2}{(\mu + \lambda_1 + \lambda_2) (\mu + \lambda_1 (1-b_2))} + \frac{\mu}{\mu + \lambda_1 + \lambda_2} - \frac{\mu}{\mu + \lambda_1 + \lambda_2} \right) e^{-\lambda_1 (v-w)} \]

\[ = 1 + \frac{\mu \lambda_1 + \lambda_1 b_2}{(\mu + \lambda_1 + \lambda_2) (\mu + \lambda_1 (1-b_2))} - \frac{\mu (\mu + \lambda_1 + \lambda_2)}{(\mu + \lambda_1 + \lambda_2) (\mu + \lambda_1 (1-b_2))} e^{-\lambda_1 (v-w)} \]

\[ = 1. \]

So, the transition probability and densities from the Service Accredited Customer set integrate as they should. The next equality to be verified is Expression (4.14). When we substitute the relevant probability and density functions in the left hand side of this expression we obtain
\[ p_{v,v\rightarrow 0,0} + \int_0^\infty \int_y^{v+\frac{b_2}{\mu}} f_{v,v\rightarrow x,y} \, dx \, dy + \int_0^v f_{v,v\rightarrow x,v} \, dx + \int_v^\infty f_{v,v\rightarrow x,x|v} \, dx = \int_0^v e^{-(\lambda_1 + \frac{\lambda_2}{b_2})} \frac{\mu}{\mu + \lambda_1 + \lambda_2} \int_y^{v+\frac{b_2}{\mu}} e^{-\lambda_1(v-x)} e^{-(\mu+\lambda_1)\frac{y-x}{b_2}} \, dx \, dy \]

So, also the transition probability and densities from the Service Unaccredited Customer set integrate as they should.
Appendix B. Check for transition probability and density functions in the two class problem
Appendix C

Working out the integrals in right hand side of $\pi(x, x)$

The expression for the limiting distribution for the stationary waiting time of unaccredited customers is given by

$$
\int_0^\infty \pi(x, x)e^{-s_1x}dx = \frac{\mu(\lambda_1 + \lambda_2)}{\mu + \lambda_1 + \lambda_2} \left( \int_0^\infty \pi(0, 0)e^{-\frac{\mu + \lambda_1(1-b_2)}{b_2}x}e^{-s_1x}dx 
+ \int_0^\infty \int_0^x \pi(v, v)e^{-\frac{\mu + \lambda_1(1-b_2)}{b_2}(x-v)}dve^{-s_1x}dx 
+ \int_0^\infty \int_x^\infty \pi(v, v)e^{-\lambda_1(v-x)}dve^{-s_1x}dx 
+ \int_0^\infty \int_0^x \int_0^\infty \pi(v, w)e^{-\frac{\mu + \lambda_1(1-b_2)}{b_2}(x-w)}dvdwe^{-s_1x}dx 
+ \int_0^\infty \int_x^\infty \int_0^\infty \pi(v, w)e^{-\lambda_1(v-w)}e^{-\lambda_2(w-x)}dvdwe^{-s_1x}dx \right).
$$

The term at the left hand side of this expression is the single laplace transform. The first term of the right hand side is also an integral which we can solve immediately. In the second term of the right hand side we recognize a convolution and by with the convolution theorem (Expression (3.11)) we can solve this term also straightforward. The third, fourth and fifth terms of the right hand side are less straightforward. Therefore, we work them out here.

1. We work out the third term of the right hand side with help of changing the order of integration. When we apply this technique we obtain
\[
\int_0^\infty \int_0^\infty \pi(v, v)e^{-(\lambda_1 + \frac{b_2}{s_2})(v-x)}dv e^{-s_1x}dx \\
= \int_0^\infty \int_0^v \pi(v, v)e^{-(\lambda_1 + \frac{b_2}{s_2})(v-x)} e^{-s_1x}dv dx \\
= \int_0^\infty \int_0^v \pi(v, v)e^{-(\lambda_1 + \frac{b_2}{s_2})v} e^{-(s_1-(\lambda_1 + \frac{b_2}{s_2}))x}dx dv \\
= \frac{1}{s_1 - (\lambda_1 + \frac{b_2}{s_2})} \left( \int_0^\infty \pi(v, v)e^{-(\lambda_1 + \frac{b_2}{s_2})v} - \int_0^\infty \pi(v, v)e^{-s_1v}dv \right) \\
= \tilde{\pi} \left( \lambda_1 + \frac{b_2}{s_2} \right) \frac{b_2}{b_2 s_1 - b_2 \lambda_1 - \lambda_2} - \tilde{\pi}(s_1) \frac{b_2}{b_2 s_1 - b_2 \lambda_1 - \lambda_2}.
\]

In this result, \( \tilde{\pi} \left( \lambda_1 + \frac{b_2}{s_2} \right) \) denotes the 1D distribution \( \tilde{\pi}(s_1) \) evaluated at the point \( s_1 = \lambda_1 + \frac{b_2}{s_2} \).

2. For the fourth term of the right-hand side we obtain

\[
\int_0^\infty e^{-s_1x} \int_0^\infty \int_0^x \int_0^w \pi(v, w)e^{-\lambda_1(v-w)}e^{-\frac{\mu+\lambda_1(1-b_2)}{b_2}(x-w)}dv dw dx \\
= \int_0^\infty \int_0^\infty \int_0^w \pi(v, w)e^{-\lambda_1(v-w)}e^{-\frac{\mu+\lambda_1(1-b_2)}{b_2}(x-w)}dv dw dx \\
= \int_0^\infty \int_0^w \pi(v, w)e^{-\lambda_1(v-w)}e^{-\frac{\mu+\lambda_1(1-b_2)}{b_2}(x-w)}dx dw \\
= \int_0^\infty \int_0^w \pi(v, w)e^{-\lambda_1(v-w)}e^{-\frac{\mu+\lambda_1(1-b_2)}{b_2}(x+w)}dx dw \\
= \int_0^\infty \int_0^w \pi(v, w)e^{-\lambda_1(v-w)}e^{-\frac{\mu+\lambda_1(1-b_2)}{b_2}(x+w)}dx dw \\
= \int_0^\infty \int_0^w \pi(v, w)e^{-\lambda_1(v-w)}e^{-\frac{\mu+\lambda_1(1-b_2)}{b_2}(x+w)}dx dw \\
= \tilde{\pi} \left( \lambda_1, s_1 - \lambda_1 \right) \frac{1}{s_1 + \frac{\mu+\lambda_1(1-b_2)}{b_2}}.
\]

In the result \( \tilde{\pi} \left( \lambda_1, s_1 - \lambda_1 \right) \) denotes the double Laplace transform \( \tilde{\pi}(s_1, s_2) \) evaluated at \( s_1 = \lambda_1 \) and \( s_2 = s_1 - \lambda_1 \).

3. And for the fifth term of the right-hand side we obtain
\[
\int_{0}^{\infty} e^{-s_1x} \int_{x}^{\infty} \int_{w}^{\infty} \pi(v, w) e^{-\lambda_1(v-w)} e^{-(\lambda_1 + \frac{s_2}{b_2})(w-x)} dwdvx
\]

\[
= \int_{0}^{\infty} \int_{0}^{w} \int_{w}^{\infty} e^{-s_1x} \pi(v, w) e^{-\lambda_1(v-w)} e^{-(\lambda_1 + \frac{s_2}{b_2})(w-x)} dvdxw
\]

\[
= \int_{0}^{\infty} \int_{0}^{w} e^{-s_1x} \pi(v, w) e^{-\lambda_1(v-w)} e^{-(\lambda_1 + \frac{s_2}{b_2})(w-x)} dxdvw
\]

\[
= \int_{0}^{\infty} \int_{w}^{\infty} e^{-(s_1-(\lambda_1+\frac{s_2}{b_2})x)} \pi(v, w) e^{-\lambda_1(v-w)} e^{-(\lambda_1+\frac{s_2}{b_2})w} dxdvw
\]

\[
= \int_{0}^{\infty} \int_{w}^{\infty} \pi(v, w) e^{-\lambda_1(v-w)} e^{-(\lambda_1+\frac{s_2}{b_2})w} dvw \left( \frac{1}{s_1 - (\lambda_1 + \frac{s_2}{b_2})} \left[ 1 - e^{-(s_1-(\lambda_1+\frac{s_2}{b_2})w)} \right] \right)
\]

\[
= \frac{1}{s_1 - (\lambda_1 + \frac{s_2}{b_2})} \int_{0}^{\infty} \int_{w}^{\infty} \left( \pi(v, w) e^{-\lambda_1v} e^{-\frac{s_2}{b_2}w} - \pi(v, w) e^{-\lambda_1v} e^{-(s_1-\lambda_1)w} \right) dvw
\]

\[
= \frac{1}{s_1 - (\lambda_1 + \frac{s_2}{b_2})} \left( \tilde{\pi} \left( \frac{\lambda_1}{b_2} - \lambda_1 \right) - \tilde{\pi} \left( \lambda_1, s_1 - \lambda_1 \right) \right)
\]

In the result \( \tilde{\pi} \left( \frac{\lambda_1}{b_2}, s_1 - \lambda_1 \right) \) denotes the double Laplace transform \( \tilde{\pi}(s_1, s_2) \) evaluated at \( s_1 = \lambda_1 \) and \( s_2 = s_1 - \lambda_1 \).
Appendix C. Working out the integrals in right hand side of $\pi(x, x)$
Appendix D

Working out the integrals in right hand side of $\pi(x, y)$

The expression for the limiting distribution for the stationary waiting time of unaccredited customers is given by

$$\int_0^\infty e^{-s_2y} \int_y^{b_2y} e^{-s_1x} \pi(x, y) dx dy = \frac{\mu \lambda_1}{b_2} \left( \int_0^\infty e^{-s_2y} \int_y^{b_2y} e^{-s_1x} \pi(0, 0) e^{\lambda_1 x} e^{-\frac{\mu + \lambda_1}{b_2^2} y} dx dy \right. $$

$$+ \int_0^\infty e^{-s_2y} \int_y^{b_2y} e^{-s_1x} \int_0^{\frac{y-b_2x}{b_2}} \pi(v, v) e^{-\lambda_1(v-x)} e^{-\frac{\mu + \lambda_1}{b_2^2} (y-v)} dv dx dy$$

$$+ \int_0^\infty e^{-s_2y} \int_y^{b_2y} e^{-s_1x} \int_0^{\frac{y-b_2x}{b_2}} \int_w^\infty \pi(v, w) e^{-\lambda_1(v-x)} e^{-\frac{\mu + \lambda_1}{b_2^2} (y-w)} dv dw dx dy$$

$$+ \left. \int_0^\infty e^{-s_2y} \int_y^{b_2y} e^{-s_1x} \int_0^{\frac{y-b_2x}{b_2}} \int_0^{\frac{w-x}{b_2}} \pi(v, w) e^{-\lambda_1(v-x)} e^{-\frac{\mu + \lambda_1}{b_2^2} (y-w)} dv dw dx dy \right).$$

The term at the left hand side of this expression is the definition of the double Laplace transform.

We work out the four terms at the right hand side of the expression as follows

1. For the first term of the right hand side of the expression we have
Appendix D. Working out the integrals in right hand side of $\pi(x, y)$

\[\int_0^\infty e^{-s_1 y} \int_y^{s_1} e^{-s_1 x} \pi(0, 0) e^{\lambda_1 x} e^{-(\mu + \lambda_1) x} \, dx \, dxdy\]
\[= \pi(0, 0) \int_0^\infty e^{-(s_2 + \frac{\mu + \lambda_1}{b_2}) y} e^{-(s_1 - \lambda_1) x} \, dx \, dxdy\]
\[= \pi(0, 0) \int_0^\infty e^{-(s_2 + \frac{\mu + \lambda_1}{b_2}) y} e^{-(s_1 - \lambda_1) x} \, dx \, dxdy\]
\[= \pi(0, 0) \frac{1}{s_1 - \lambda_1} \int_0^\infty e^{-(s_1 - \lambda_1 + \frac{\mu + \lambda_1}{b_2}) y} e^{-(\frac{\lambda_1}{b_2} + s_2 + \frac{\mu + \lambda_1}{b_2}) y} \, dy\]
\[= \pi(0, 0) \frac{1}{s_1 - \lambda_1} \int_0^\infty e^{-(s_1 - \lambda_1 + \frac{\mu + \lambda_1}{b_2}) y} e^{-(\frac{\lambda_1}{b_2} + s_2 + \frac{\mu + \lambda_1}{b_2}) y} \, dy\]
\[= \pi(0, 0) \frac{1}{s_1 - \lambda_1} \left[ \frac{1}{s_1 + s_2 + \mu + \lambda_1(1-b_2)} - \frac{1}{s_1 + s_2 + \lambda_1} \right] \]
\[= \pi(0, 0) \frac{b_2(1-b_2)(s_1 + b_2 s_2 + \mu)}{b_2(1-b_2)} \]
\[= \pi(0, 0) \frac{(s_1 + s_2)(1) + \mu + \lambda_1(1-b_2))(s_1 + b_2 s_2 + \mu)}{b_2(1-b_2)} \]

2. For the second term of the right hand side of the expression we have

\[\int_0^\infty e^{-s_2 y} \int_y^{s_1} e^{-s_1 x} \int_0^{y b_2} \pi(v, v) e^{-\lambda_1(v-x)} e^{-\frac{\mu + \lambda_1}{b_2} (v-y)} \, dxdy \, dxdv \]
\[= \int_0^\infty \int_y^{s_1} e^{-s_2 y} \pi(v, v) e^{-\lambda_1(v-x)} e^{-\frac{\mu + \lambda_1}{b_2} (v-y)} \, dxdy \, dxdv \]
\[= \int_0^\infty \pi(v, v) e^{-\frac{\mu + \lambda_1(1-b_2)}{b_2} v} \int_0^{v - \frac{1}{b_2}} e^{-(s_2 + \frac{\mu + \lambda_1}{b_2}) y} \int_y^{v - \frac{1}{b_2}} e^{(\lambda_1 - s_1) v} \, dxdy \, dxdv \]
\[= \int_0^\infty \pi(v, v) e^{-\frac{\mu + \lambda_1(1-b_2)}{b_2} v} \int_0^{v - \frac{1}{b_2}} e^{-(s_2 + \frac{\mu + \lambda_1}{b_2}) y} \int_y^{v - \frac{1}{b_2}} e^{(\lambda_1 - s_1) v} \, dxdy \, dxdv \]
\[= \int_0^\infty \pi(v, v) e^{-\frac{\mu + \lambda_1(1-b_2)}{b_2} v} \int_0^{v - \frac{1}{b_2}} e^{-(s_2 + \frac{\mu + \lambda_1}{b_2}) y} \int_y^{v - \frac{1}{b_2}} e^{(\lambda_1 - s_1) v} \, dxdy \, dxdv \]
\[= \int_0^\infty \pi(v, v) e^{-\frac{\mu + \lambda_1(1-b_2)}{b_2} v} \int_0^{v - \frac{1}{b_2}} e^{-(s_2 + \frac{\mu + \lambda_1}{b_2}) y} \int_y^{v - \frac{1}{b_2}} e^{(\lambda_1 - s_1) v} \, dxdy \, dxdv \]
\[= \int_0^\infty \pi(v, v) e^{-\frac{\mu + \lambda_1(1-b_2)}{b_2} v} \int_0^{v - \frac{1}{b_2}} e^{-(s_2 + \frac{\mu + \lambda_1}{b_2}) y} \int_y^{v - \frac{1}{b_2}} e^{(\lambda_1 - s_1) v} \, dxdy \, dxdv \]
\[= \pi(0, 0) \frac{(s_1 + s_2) b_2(1-b_2)}{b_2(1-b_2)} \]
\[= \pi(0, 0) \frac{(s_1 + s_2) b_2(1-b_2)}{b_2(1-b_2)} \]

In the result $\tilde{\pi}(s_1 + s_2)$ denotes the Laplace transform $\tilde{\pi}(s_1)$ evaluated at $s_1 = s_1 + s_2$.

3. For the third term of the right hand side of the expression we have
\[
\int_0^\infty e^{-s_2y} \int_y^{y/2} e^{-s_1x} \int_0^{y-x/2} \int_w^{y-x/2} \pi(v, w)e^{-\lambda_1(v-x)}e^{-\frac{\mu+\lambda_1}{b_2}(y-w)}
\]
\[
\times \frac{dvdw}{b_2(1-b_2)}
\]
\[
= \int_0^\infty \int_0^y \int_y^{y-(1-b_2)w} \int_w^{y-(1-b_2)w} e^{-s_2y}e^{-s_1x} \pi(v, w)e^{-\lambda_1(v-x)}e^{-\frac{\mu+\lambda_1}{b_2}(y-w)}
\]
\[
\times \frac{dvdw}{b_2(1-b_2)}
\]
\[
= \int_0^\infty \int_0^\infty \int_0^{y-(1-b_2)w} \int_w^{y-(1-b_2)w} e^{-s_2y}e^{-s_1x} \pi(v, w)e^{-\lambda_1(v-x)}e^{-\frac{\mu+\lambda_1}{b_2}(y-w)}
\]
\[
\times \frac{dvdw}{b_2(1-b_2)}
\]
\[
= \int_0^\infty \int_0^{\infty} \int_0^{\infty} \int_0^{\infty} \int_0^{\infty} \int_0^{\infty} \int_0^{\infty} \int_0^{\infty} \int_0^{\infty} \int_0^{\infty} \pi(v, w)e^{-\lambda_1v}e^{-\frac{\mu+\lambda_1}{b_2}w}
\]
\[
\times \int_0^{\infty} e^{-(s_2+\frac{\mu+\lambda_1}{b_2})y} \int_y^{y-(1-b_2)w} e^{-\lambda_1y}e^{-\frac{\mu+\lambda_1}{b_2}(y-w)}
\]
\[
\times \frac{dvdw}{b_2(1-b_2)}
\]
\[
= \pi(\lambda_1, s_1 + s_2 - \lambda_1)
\]
\[
\times \frac{(b_2(s_1 + s_2) + \mu + \lambda_1(1-b_2))(s_1 + b_2s_2 + \mu)}{b_2(1-b_2)}
\]

In the result \(\tilde{\pi}(\lambda_1, s_1 + s_2 - \lambda_1)\) denotes the Laplace transform \(\tilde{\pi}(s_1, s_2)\) evaluated at \(s_1 = \lambda_1\) and \(s_2 = s_1 + s_2 - \lambda_1\).

4. For the fourth term of the right hand side of the expression we have
Appendix D. Working out the integrals in right hand side of $\pi(x, y)$

\[
\int_0^\infty e^{-s_2 y} \int_y^{t_2} e^{-s_1 x} \int_0^{\infty} \int_{x-w}^{\infty} \pi(v, w) e^{-\lambda_1(v-x)} e^{-\frac{\mu + \lambda_1}{\mu} (y-w)} dvwdxdy
\]

\[
= \int_0^\infty \int_0^{y} \int_{y-w}^{\infty} \int_{x-w}^{\infty} e^{-s_2 y} e^{-s_1 x} \pi(v, w) e^{-\lambda_1(v-x)} e^{-\frac{\mu + \lambda_1}{\mu} (y-w)} dvwdxdwy
\]

\[
= \int_0^\infty \int_0^{\infty} \int_{y-w}^{\infty} \int_{x-w}^{\infty} e^{-s_2 y} e^{-s_1 x} \pi(v, w) e^{-\lambda_1(v-x)} e^{-\frac{\mu + \lambda_1}{\mu} (y-w)} dvwdxdydw
\]

\[
= \int_0^\infty \int_0^{\infty} \int_{y-w}^{\infty} \int_{x-w}^{\infty} e^{-s_2 y} e^{-s_1 x} \pi(v, w) e^{-\lambda_1(v-x)} e^{-\frac{\mu + \lambda_1}{\mu} (y-w)} dx dy dw dv
\]

\[
= \int_0^\infty \int_0^{\infty} \int_{y-w}^{\infty} \int_{x-w}^{\infty} \pi(v, w) e^{-\lambda_1(v-x)} e^{-\frac{\mu + \lambda_1}{\mu} (y-w)} dvwdx dy dw
\]

\[
= \int_0^\infty \int_0^{\infty} \int_{y-w}^{\infty} \int_{x-w}^{\infty} \pi(v, w) e^{-\lambda_1(v-x)} e^{-\frac{\mu + \lambda_1}{\mu} (y-w)} dvwdx dy dw
\]

\[
= (\tilde{\pi}(\lambda_1, s_1 + s_2 - \lambda_1) - \tilde{\pi}(s_1, s_2)) \frac{b_2}{(s_1 + b_2 s_2 + \mu)}
\]

In the result $\tilde{\pi}(\lambda_1, s_1 + s_2 - \lambda_1)$ denotes the Laplace transform $\tilde{\pi}(s_1, s_2)$ evaluated at $s_1 = \lambda_1$ and $s_2 = s_1 + s_2 - \lambda_1$. 
Appendix E

Matlab code: simulation file

% Simulation of the maximum priority process for the two class case.

% Written by: D.J.J. Dams
% Date: 30/11/2012
% Location: University of Melbourne

% This m-file contains the program for simulating the maximum priority process.
% The queue that is investigated is the M/M/1 queue.
% Thus poisson arrivals and service times drawn from an exponential distribution.
% We consider a queue having two types of classes, i.e. class 1 and class 2 customers, having their own priority rate, b1 and b2 respectively. It is assumed that both classes have the same service time distribution.

clear all; close all; clc;

%%% DECLARATIONS

% Total number of customer classes:
N = 2;

% Boolean to indicate whether system is in busy period or not. Initially this is not true, because system has no customers:
bBusyPeriod = 0;

% Arrival rates of the different classes:
lambda(1) = 1; % [arrivals/time unit]
lambda(2) = 2; % [arrivals/time unit]
lambda_tot = sum(lambda); % [arrivals/time unit]

% Priority rates with as many entries as classes
b(1) = 1; % [1/time unit]
Appendix E. Matlab code: simulation file

\[
\text{b(2)} = 0.5; \quad \% \text{[1/time unit]}
\]

% Total number of customers entering the system during time interval
% \([0, \inf )\):
% TotalCustomers = 5;
% TotalCustomers = 20;
% TotalCustomers = 100;
TotalCustomers = 10000;

% Print information to screen:
string_classes = ['The number of classes is, N = ', num2str(N)];
string_lambdas = ['The arrival rates are, lambda_1 = ', num2str(lambda(1)), ...
\quad \text{ and lambda}_2 = ', num2str(lambda(2))];
string_prios = ['The priority rates are, b_1 = ', num2str(b(1)), ...
\quad \text{ and b}_2 = ', num2str(b(2))];
string_customers = ['The total number of customers entering during simulation = '
\quad \text{num2str(TotalCustomers)}];

disp('System information: ')
disp(string_classes)
disp(string_lambdas)
disp(string_prios)
disp(string_customers)

\% Service time distribution
\% Distribution for service times of both classes, the assumption is made that
\% the service time distributions are the same for class 1 and 2 customers.

% Service times are distributed according to an exponential distribution:
mu = 4; \quad \% \text{server rate [customers/time unit]}
MeanServiceDist = 1/mu; \quad \% \text{mean of service time distribution}

% Check utilization of the system:
rho = lambda_tot/mu;
if rho < 1
    disp('utilization = ok, less than 1 -> mu is okay')
disp("")
else
    disp('utilization = error, more than 1 -> mu is wrong, CHANGE MU')
disp("")
end

\% Preallocation of memory
tau = zeros(1,TotalCustomers); \quad \% \text{inter-arrival times}
T_Arrival = zeros(1,TotalCustomers); \quad \% \text{arrival times at SERVER}
X = zeros(1,TotalCustomers+1); \quad \% \text{service times}
D = zeros(1,TotalCustomers); \quad \% \text{departure times from SERVER}
AccreditationLevel = zeros(1,TotalCustomers+1); \quad \% \text{accreditation level}
\[ E = \text{zeros}(\text{TotalCustomers}, 2); \]

% The array Time and matrix M cannot be preallocated because they are not % always the same. It depends on the number of busy periods that occur % during simulation. The array and matrix are initialized here. It is known % that in the origin, thus \( t = 0 \) the maximal priorities \( M_1 \) and \( M_2 \) are equal % to 0.

% The array Time is used to store the time points that are of importance. % i.e. the time points when a customer arrives or departs from the server. % The departure of the \( m \)th customer is equal to the arrival of the % \((m+1)\)th customer to the system IF the system is in a busy period. % When the system is not in a busy period, thus no customer is in the % queue when the \( m \)th customer departs from the server, the departure time % of the \( m \)th customer is stored in the array Time. % Note: the time points that are of importance are stored in duplicate in % the array. The reason therefore becomes clear when the maximal % priority matrix is explained.

\[ \text{Time} = [ 0, 0 ]; \]

% The matrix M is used to store the maximal priorities at the time points % stored in the array Time. The first row contains the priorities belonging % to the process \( M_1 \) and the second row contains the priorities belonging to % the process \( M_2 \). % When a customer leaves the system a jump in maximal priority is observed. % This jump occurs at one time point. The array Time contains duplicates of % every time point for this reason. Every column of the matrix M is related % to the column of the array Time. % Thus: \[ M = \begin{bmatrix} M_1(0) & M_1(0) & M_1(T_1) & M_1(T_1) & M_1(D_1-) & M_1(D_1) & \ldots & M_1(D_N-) & M_1(D_N) \\ M_2(0) & M_2(0) & M_2(T_1) & M_2(T_1) & M_2(D_1-) & M_2(D_1) & \ldots & M_2(D_N-) & M_2(D_N) \end{bmatrix} \]

\[ M = [ 0, 0 ; 0, 0 ]; \]

% To keep track of the total number of busy periods the system went through % a counter for the amount of busy periods is initialized:

\[ \text{BusyPeriodCounter} = 0; \]

% ---------------Customer_class_at_accredetation_level-------------------------- % If a customer initiates a busy period, the probability that this customer % is of class 1 = \( \lambda(1)/(\lambda_{\text{tot}}) \)
% % If a served customer is served at accreditation level 1, this customer % is of class 1, with probability:
% \[ - \text{P(class 1)} = 1 \]
% % If a served customer is served at accreditation level 2, this customer is % of class 1 or 2, with probabilities:
% \[ - \text{P(class 1)} = (\lambda(1)/b(1))/(\lambda(1)/b(1) + \lambda(2)/b(2)) \]
% \[ - \text{P(class 2)} = (\lambda(2)/b(2))/(\lambda(1)/b(1) + \lambda(2)/b(2)) \]
% Calculation of probabilities if customer initiates busy period:
P1_initbp = lambda(1)/lambda_tot;
P2_initbp = 1-P1_initbp;

string1 = ['If a customer initiates a busy period, this customer is of class 1 with P: ', num2str(P1_initbp), ' and class 2 with P: ', num2str(P2_initbp)];

% Calculation of probabilities if accreditation level 2:
P1_acc2 = (lambda(1)/b(1))/(lambda(1)/b(1) + lambda(2)/b(2));
P2_acc2 = (lambda(2)/b(2))/(lambda(1)/b(1) + lambda(2)/b(2));

string2 = ['If accreditation level is 2 the customer is of class 1 with P: ', num2str(P1_acc2), ' and class 2 with P: ', num2str(P2_acc2)];

% Print information to screen:
disp(string1)
disp('If accreditation level is 1 the customer is of class 1')
disp(string2)
disp('  ')

%SIMULATION

for m = 1:TotalCustomers
    if bBusyPeriod == 0
        tau(m) = exprnd(1/lambda_tot); % inter-arrival time of m'th customer
        Time = [Time, T_Arrival(m), T_Arrival(m)];
        T_Arrival(m) = Time(1, length(Time)) + tau(m);
    end
end
When a customer enters the system and the system is not in a busy period the maximal priority is equal to 0 because there is no accumulated priority.

Maximal priorities belonging to time point stored in matrix M:

\[
M = \begin{bmatrix}
0 & 0 \\
0 & 0
\end{bmatrix};
\]

Determination of service time of m\textsuperscript{th} arriving customer

Service time of m\textsuperscript{th} arriving customer from exponential distribution:

\[X(m) = \text{exprnd}(\text{MeanServiceDist});\]

Determination of departure time of m\textsuperscript{th} customer

The departure time of the m\textsuperscript{th} customer is equal to \( T_{\text{Arrival}}(m) + X(m) \) since the customer initiates a busy period, thus server is idle.

\[D(m) = T_{\text{Arrival}}(m) + X(m);\]

If the busy period continues than \( D(m) = T(m+1) \). Time point stored in array Time:

\[\text{Time} = [\text{Time}, D(m), D(m)];\]

Determination of maximal priorities from m\textsuperscript{th} arrival

Maximal priorities belonging to time point stored in matrix M:

\[
M = \begin{bmatrix}
(b(1) \times X(m)); & \% \text{M1 at } D_m \\
(b(2) \times X(m)) & \% \text{M2 at } D_m
\end{bmatrix};
\]

elseif bBusyPeriod == 1

Determination of arrival time of the customer

Departure time previous customer (m−1)\textsuperscript{st} is known, this departure time is equal to the arrival time at the server of the m\textsuperscript{th} customer:

\[T_{\text{Arrival}}(m) = \text{Time}(1, \text{length(Time)});\]

Service time of m\textsuperscript{th} customer is known. Is determined before in lines 287 or 316.

Determination of departure time of the customer

The departure time of the m\textsuperscript{th} customer is equal to:

\[D(m) = T_{\text{Arrival}}(m) + X(m);\]

Store D(m) in vector Time. If the busy period continues than \( D(m) = T(m+1) \):

\[\text{Time} = [\text{Time}, D(m), D(m)];\]

Determination of accumulated max priority of the customer

Maximal priorities belonging to time point stored in matrix M:

\[
M = \begin{bmatrix}
(M(1, \text{length(M)}) + b(1) \times X(m)); & \% \text{M1 at } D_m \\
(M(2, \text{length(M)}) + b(2) \times X(m)) & \% \text{M2 at } D_m
\end{bmatrix};
\]
end

% The maximal priorities at Dm are known. Now, the priorities at the
% time points Dm should be obtained. This priority depends on the which
% type of customer is served. In the two class case the different
% possibilities are:
% − Customer accredited at level 1 is served, this is always a
%   customer of class 1.
% − Customer accredited at level 2 is served, this could be
%   either a customer of class 1 or 2.
%
% There is a customer in the system thus a busy period is initialized.
% Now, it must be investigated if the system stays in a busy period
% after serving a customer at accreditation level 1 or 2 or if the busy
% period ends.

%-------------At_the_m\_th_service_completion_time_with_m_\_>=_1-------------
% Draw E(m,1) from exponential distribution with mean b(1)/lambda(1):
E(m,1) = exprnd(b(1)/lambda(1)); % accreditation level k = 1

if E(m,1) < ( M(1,length(M)) - M(2,length(M) ) )
   % bBusyPeriod (becomes) true:
   bBusyPeriod = 1;

   %-------------Determination_of_accumulated_max_priority_of_the_customer-----
   % Maximal priorities belonging to time point stored in matrix M:
   M = [M, [M(1,length(M)) - E(m,1); % M1 at Dm
            M(2,length(M))   ]] ; % M2 at Dm

%-------------Determination_of_service_time_of_next_customer-------------
% Service time of (m+1)th arriving customer is determined:
   X(m+1) = exprnd(MeanServiceDist);
else if E(m,1) > ( M(1,length(M)) - M(2,length(M) ) )
    % Draw E(m,2) from exponential distribution with mean
    % inv((lambda(1)/b(1)) + (lambda(2)/b(2))):
    E(m,2) = exprnd(inv((lambda(1)/b(1))+(lambda(2)/b(2)))); % accreditation

    if E(m,2) < ( M(2,length(M) ) - 0 )
        % bBusyPeriod (becomes) true:
        bBusyPeriod = 1;

        %-------------Determination_of_accumulated_max_priority_of_the_customer-----
        % Maximal priorities belonging to time point stored in matrix M:
        M = [M, [M(2,length(M)) - E(m,2);
                  M(2,length(M)) - E(m,2)]];
% Determination_of_service_time_of_next_customer
% Service time of (m+1)th arriving customer is determined:
X(m+1) = exprnd(MeanServiceDist);

elseif E(m,2) > (M(2, length(M)) - 0)
% Busy period has finished. There are no customers waiting in
% the queue before the server.

% Maximal priorities belonging to time point stored in matrix M:
M = [M, [0;
     0]];

% Busy period becomes/stays false:
bBusyPeriod = 0;

% Busy period counter is increased:
BusyPeriodCounter = BusyPeriodCounter + 1;

end
end
end

% Plot

PastBusyPeriods = BusyPeriodCounter

% Delete first two columns of both the array Time and matrix M:
Time = Time(1,3:length(Time));
M = M(:,3:length(M));

% Split matrix M in M_Dmin and M_D
for z = 1 : length(M)
    if mod(z,2) == 1 % oneven: D-
        M_Dmin(:,(z/2+.5)) = M(:,z);
    elseif mod(z,1) == 0 % even: D
        M_D(:,(z/2)) = M(:,z);
    end
end
% Plot the maximum priority process
figure
plot(Time,M(1,:),’-b’)
hold on
plot(Time,M(2,:),’-r’)
legend(’M1’,’M2’,’Location’,’NorthWest’)
xlabel(’time ’)
ylabel(’maximal accumulated priority ’)
title(’Maximal priorities versus time’)

% Plot the (M2,M1) plot
figure
plot(M_D(1,:),M_D(2,:),’*b’)
hold on
plot(M_D(1,:),M_D(1,:),’-r’)
plot(M_D(1,:),b(2)/b(1))*M_D(1,:),’-g’)
legend(’Accumulated priorities’,’M_2(t; t\cdot D) = M_1(t; t\cdot D)’,’M_2(t; t\cdot D) = (b_2/b_1)\cdot M_1(t; t\cdot D)’)
xlabel(’M_1(t; t\cdot D)’)
ylabel(’M_2(t; t\cdot D)’)
title(’Matrix M_{\{D\}}’)

Appendix E. Matlab code: simulation file
Appendix F

Matlab code: determining waiting times customers

% M-file for determining the waiting times of the different customers
function [W_accr,W_unaccr,W_idle]= WaitingTimesCustomers(M,b1,b2)

%-------INPUT_FOR_PROGRAM--------------------------------------------------------------------------------------
% Input:
% M matrix from Simulation program of the maximal priority process.
% Priority rate class 1 customers: b1=1.
% Priority rate class 2 customers: b2=.5.

%-------PREALLOCATION_OF_MEMORY----------------------------------------------------------------------------------
Prio_D = zeros(2,length(M)/2); % Matrix to store prioritys at D
% The vectors W_accr and W_unaccr cannot be preallocated because they are not
% always the same. They are dependent on the number of busy periods and the
% probability of serving accredited and unaccredited customers.
W_accr = [0]; % Vector to store waiting times of accredited customers
W_unaccr = [0]; % Vector to store waiting times of unaccredited customers
W_idle = [0];

%-------WAITING_TIME_DETERMINATION--------------------------------------------------------------------------------
% We are interested in the priorities at D and not at D−. This means that
% we have to cut some data from the matrix M. We are only interested in the
% even entries of the matrix. These are stored in matrix Prio_D
for i = 1:length(M)
    if mod(i,2) == 0
        entry = i/2;
        Prio_D(:,entry) = M(:,i);
    end
end
% Now we need to evaluate the matrix Prio_D. We know that M1 is always
% greater or equal to M2. If M1 is smaller than M2 the code contains and
% error and the program is not correct.
% Furthermore, we can distinguish two cases:
% If M1 > M2: we know that an accredited customer is going into service.
% Its waiting time is equal to M1/b1.
% If M1 = M2: we know that an unaccredited customer is going into service.
% Its waiting time is equal to M2/b2.

for j = 1:length(Prio_D)
    if Prio_D(1,j) > Prio_D(2,j)
        % The waiting time of the accredited customers
        W_accr = [ W_accr, Prio_D(1,j) ];
    elseif Prio_D(1,j) == Prio_D(2,j) && Prio_D(2,j) > 0
        % The waiting time of the unaccredited customers
        W_unaccr = [ W_unaccr, Prio_D(2,j) ];
    elseif Prio_D(1,j) == Prio_D(2,j) && or(Prio_D(1,j) == 0, Prio_D(2,j) == 0)
        % Customer enters service in idle period
        W_idle = [ W_idle, Prio_D(1,j) ];
    else
        disp('M1 < M2, is not possible, CHECK PROGRAM')
    end
end

W_accr = W_accr(2:length(W_accr));
W_unaccr = W_unaccr(2:length(W_unaccr))./b2;
W_idle;
end
Appendix G

Matlab code: probability density file

% M-file for plotting a PDF of waiting times of the accredited and unaccredited customers that we obtain from the simulation of the two class accumulating priority queue.
close all; clear all; clc;

%-----------------------SIMULATION_PDF---------------------------------
% Load data:
% Waiting times from simulation with 10000 customers
load WaitingTimesAccredited10000 % Waiting_Accredited
load WaitingTimesUnaccredited10000 % Waiting_Unaccredited

% Number of bins for the histogram
nbin1 = 35; % used for accredited customers
nbin2 = 100; % used for unaccredited customers

% Creating histogram
[FreqAccr TimeAccr] = hist(Waiting_Accredited,nbin1);
[FreqUnaccr TimeUnacr] = hist(Waiting_Unaccredited,nbin2);

% To create the probability density we integrate the obtained histograms
Area_Accr = trapz(TimeAccr,FreqAccr);
Area_Unaccr = trapz(TimeUnaccr,FreqUnaccr);

% Divide the entries of the histogram by their areas
FreqAccr_NORM = FreqAccr./Area_Accr;
FreqUnacr_NORM = FreqUnacr./Area_Unaccr;

% Check if area probability density is equal to one
Area_PDF_Accr = trapz(TimeAccr,FreqAccr_INT)
Area_PDF_Unaccr = trapz(TimeUnaccred,FreqUnaccred_INT)
% PLOT
% Plot for Probability Density Accredited Customers
figure
hold on
plot(TimeAccr,FreqAccr_NORM,'or')
plot(TimeAccr,FreqAccr_NORM,'—k')
xlabel('Time')
ylabel('Probability Density')
title('PDF Waiting Time for Accredited Customers')
axis([0 5 0 1.1])

% Plot for Probability Density Unaccredited Customers
figure
hold on
plot(TimeUnaccr,FreqUnaccr_NORM,'or')
plot(TimeUnaccr,FreqUnaccr_NORM,'—k')
xlabel('Time')
ylabel('Probability Density')
title('PDF Waiting Time for Unaccredited Customers')
axis([0 10 0 1.1])
Appendix H

Matlab code: Euler method algorithm

The Euler method algorithm as described in Section 5.2.1 is given by

```matlab
% Numerical Inversion of Laplace Transforms of Probability Distributions
% The Euler method

clear all; close all; clc;

% Making an array with time points
Tbegin = 10^(-8);
Tend = 10;
Tpoints = 10000;
T = linspace(Tbegin,Tend,Tpoints);

% METHOD 1: EULER
% Parameters
M = 11; % then there should be 12 binomial coefficients
Ntr = 15; % makes the computation a sum of 27 terms
A = 8*log(10); % 18.4, Discretization error 10^(-8)

SU = zeros(Tpoints,M+2);
C = zeros(1,M+1);
Sum = zeros(1,Tpoints);

Avgsu = zeros(1,Tpoints);
Avgsul = zeros(1,Tpoints);
Fun = zeros(1,Tpoints);
Fun1 = zeros(1,Tpoints);
Errt = zeros(1,Tpoints);

% Binomial Coefficients
for m = 1:M+1
```

83
C(m) = nchoosek(M,m-1);
end

for z = 1:Tpoints
    U = exp(A/2)/T(z);
    X = A/(2*T(z));
    H = pi/T(z);

    % Summation part of expression for s_n(t)
    Sum(z) = fnRf(X,0)/2; % left part of equation 13
    for n = 1 : Ntr
        Y = n*H;
        Sum(z) = ((-1)^n)*fnRf(X,Y) + Sum(z);
    end

    % S_n+k
    SU(z,1) = Sum(z);
    for k = 1 : M+1
        N = Ntr + k;
        Y = N*H;
        SU(z,k+1) = SU(z,k) + ((-1)^N)*fnRf(X,Y);
    end

    % Expression for E(m,n,t)
    Avgsu(z) = 0;
    Avgsu1(z) = 0;
    for j = 1:M+1
        Avgsu(z) = C(j)*SU(z,j) + Avgsu(z);
        Avgsu1(z) = C(j)*SU(z,j+1) + Avgsu1(z);
    end

    Fun(z) = U*Avgsu(z)/2048;
    Fun1(z) = U*Avgsu1(z)/2048;

    Errt(z) = abs(Fun(z) - Fun1(z))/2;
end

TIME = T;
FUNCTION_Euler = Fun1;
Truncation_Error_Estimate_Euler = Errt;

As can be seen in the code a function file (fnRf.m) is used in the algorithm. In this function file we define the function in the Laplace domain we like to invert. The function file is given by

function Rfs = fnRf(X,Y)
    % Defining the dummy variable S
    S = complex(X,Y);

    % Function in the Laplace domain, for example
    Fs = 1/(1+S);
% Output
Rfs = real(Fs);
end
Appendix H. Matlab code: Euler method algorithm
Appendix I

Matlab code: Post-Widder method algorithm

The Post-Widder method algorithm as described in Section 5.2.2 is given by

```
% Numerical Inversion of Laplace Transforms of Probability Distributions
% The Post–Widder method

clear all; close all; clc;

% Making an array with time points
Tbegin = 10ˆ(-8);
Tend = 10;
Tpoints = 10000;
T = linspace(Tbegin,Tend,Tpoints);

% METHOD 2: POST WIDDER

% Parameters
NN = 6;
G = zeros(Tpoints,NN);
Sum2 = zeros(1,Tpoints);
Fun2 = zeros(1,Tpoints);

for z= 1:Tpoints
    for i = 1:NN
        N = 10*i;
        E = 8;
        R = (1/10)*(E/(2*N));
        U = (N+1)/(T(z)*2*N*(R^N));
        H = pi/N;

        Sum2(z) = 0;
        for j = 1:N-1

```

87
Appendix I. Matlab code: Post-Widder method algorithm

\[
S = (N+1) \ast (1-R \ast \exp (\text{complex}(0, H \ast j))) / T(z);
\]
\[
\text{Sum2}(z) = \text{Sum2}(z) + ((-1)^{\ast j} \ast \text{fnF}(S));
\]
end

\[
\text{Sum2}(z) = \text{fnF}((N+1)\ast(1-R)/T(z)) + 2\ast\text{Sum2}(z) + (-1)^N \ast \text{fnF}((N+1)\ast(1+R)/T(z));
\]
G(z,i) = U*Sum2(z);
end

Fun2(z) = 0;
for k = 1:NN
\[
Wt = ((-1)^{(NN-k)}) \ast (k \ast \text{factorial}(k) \ast \text{factorial}(NN-k));
\]
Fun2(z) = Fun2(z) + Wt*G(z,k);
end
end
TIME = T;
FUNCTION PostWidder = Fun2;

As can be seen in the code a function file (fnF.m) is used in the algorithm. In this function file we define the function in the Laplace domain we like to invert. The function file is given by

function reFs = fnF(S)
    % Function in the Laplace domain, for example
    Fs = 1/(1+S);

    % Output
    reFs = real(Fs);
end