# An Application of Matrix Analytic Methods to Queueing Models with Polling

by

Kevin Granville

A thesis

presented to the University of Waterloo

in fulfilment of the

Doctor of Philosophy

in

Statistics

Waterloo, Ontario, Canada, 2019

© Kevin Granville 2019

# Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

| | |
|---|---|
| External Examiner | Dr. Yiqiang Zhao<br>Associate Dean (Research and Graduate Studies),<br>Faculty of Science, Carleton University |
| Supervisor | Dr. Steve Drekic<br>Professor, Statistics and Actuarial Science |
| Internal Members | Dr. Gordon Willmot<br>Professor, Statistics and Actuarial Science |
| | Dr. Yi Shen<br>Assistant Professor, Statistics and Actuarial Science |
| Internal-external Member | Dr. Qi-Ming He<br>Professor, Management Sciences |

# Author's Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

# Abstract

We review what it means to model a queueing system, and highlight several components of interest which govern the behaviour of customers, as well as the server(s) who tend to them. Our primary focus is on polling systems, which involve one or more servers who must serve multiple queues of customers according to their service policy, which is made up of an overall polling order, and a service discipline defined at each queue. The most common polling orders and service disciplines are discussed, and some examples are given to demonstrate their use. Classic matrix analytic method theory is built up and illustrated on models of increasing complexity, to provide context for the analyses of later chapters. The original research contained within this thesis is divided into two halves, finite population maintenance models and infinite population cyclic polling models.

In the first half, we investigate a 2-class maintenance system with a single server, expressed as a polling model. In Chapter 2, the model we study considers a total of $C$ machines which are at risk of failing when working. Depending on the failure that a machine experiences, it is sorted into either the class-1 or class-2 queue where it awaits service among other machines suffering from similar failures. The possible service policies that are considered include exhaustive, non-preemptive priority, and preemptive resume priority. In Chapter 3, this model is generalized to allow for a maintenance float of $f$ spare machines that can be turned on to replace a failed machine. Additionally, the possible server behaviours are greatly generalized. In both chapters, among other topics, we discuss the optimization of server behaviour as well as the limiting number of working machines as we let $C \to \infty$. As these are systems with a finite population (for a given $C$ and $f$), their steady-state distributions can be solved for using the algorithm for level-dependent quasi-birth-and-death processes without loss of accuracy.

When a class of customers are impatient, the algorithms covered in this thesis require their queue length to be truncated in order for us to approximate the steady-state distribution for all but the simplest model. In Chapter 4, we model a 2-queue polling system with impatient customers and $k_i$-limited service disciplines. Finite buffers are assumed for both queues, such that if a customer arrives to find their queue full then they are blocked and lost forever. Finite buffers are a way to interpret a necessary truncation level, since we can simply assume that it is impossible to observe the removed states. However, if we are interested in approximating an infinite buffer system, this inconsistency will bias the steady-state probabilities if blocking probabilities are not negligible. In Chapter 5, we introduce the Unobserved Waiting Customer approximation as a way to reduce this natural biasing that is incurred when approximating an infinite buffer system. Among the queues considered within this chapter is a $N$-queue system with exhaustive service and customers who may or may not be impatient. In Chapter 6, we extend this approximation to allow for reneging rates that depend on a customer's place in their queue. This is applied to a $N$-queue polling system which generalizes the model of Chapter 4.

# Acknowledgements

I would like to begin by thanking my PhD supervisor, Dr. Steve Drekic. Over the past five years he has been a mentor, colleague, and friend, who has continually supported me as I developed my skills both as a researcher and as an educator. Ever since introducing me to the world of queueing theory, he has always encouraged me to follow my instincts and tackle research problems of my own design, as well as providing me many opportunities to present my work to peers in our field. I am extremely grateful to have had the privilege to work with him, as well as for the time and effort that he has spent helping me get to where I am today.

I would also like to thank my Master's supervisor Dr. Adam Kolkiewicz and Bachelor's supervisor Dr. Zhaozhi Fan for helping guide me through these degrees, for pushing me to pursue further work in academics, as well as helping set the foundations of my experience with research as a younger student.

I am grateful to the members of my thesis committee, Dr. Yiqiang Zhao, Dr. Qi-Ming He, Dr. Gordon Willmot, and Dr. Yi Shen. In addition to helping with this process, I am thankful to Dr. He for inviting me to participate in a series of workshops, as well as to Dr. Willmot and Dr. Shen for having taught me courses on ruin theory and stochastic processes, respectively.

I am thankful for the assistance of Mary Lou Dufton who was my first contact in the Department of Statistics and Actuarial Science. In addition to administrative support, she helped in my decision to attend the University of Waterloo and in the selection of both of my graduate programs, while also being the one who introduced me to Dr. Drekic.

I would like to express my gratitude to my friends and family. To the friends who I have known most of my life, as well as those who I met in Waterloo and helped make my time as a graduate student enjoyable. To my sister, my brother, and my grandparents. Finally, I would like to thank my parents, whose unwavering love and support have allowed me to pursue my dreams and find my place in life. Thank you, everyone.

# Dedication

I dedicate this thesis to my loving parents, Scott and Marilyn.

# Contents

# List of Figures

# List of Tables

# List of Abbreviations

| | |
|---|---|
| $BP$ | Busy Period (random variable) |
| $BP_C$ | Level-$C$ Busy Period (random variable) |
| $\mathrm{C}_k^f$ | Coxian-$k$ with Feedback (distribution) |
| CDF | Cumulative Distribution Function |
| CTMC | Continuous-Time Markov Chain |
| DTMC | Discrete-Time Markov Chain |
| $\mathrm{E}_k$ | Erlang-$k$ (distribution) |
| $\mathrm{E}_k^f$ | Erlang-$k$ with Feedback (distribution) |
| Exp | Exponential (distribution) |
| FB | Finite Buffer |
| FCFS | First-Come, First-Served |
| G | General (distribution, Kendall's notation) |
| $\mathrm{H}_k$ | Hyperexponential-$k$ (distribution) |
| IB | Infinite Buffer |
| KBE | Kolmogorov Backward Equations |
| KFE | Kolmogorov Forward Equations |
| LN | Log-Normal (distribution) |
| LST | Laplace-Stieltjes Transform |
| M | Memoryless (Kendall's notation) |
| MAM | Matrix Analytic Methods |
| MVA | Mean Value Analysis |
| NP | Non-Preemptive Priority (service policy) |
| P | Preemptive Resume Priority (service policy) |
| PASTA | Poisson Arrivals See Time Averages |
| PDF | Probability Density Function |
| PH | Phase-type (distribution, Kendall's notation) |
| PMF | Probability Mass Function |
| PSA | Power-Series Algorithm |
| QBD | Quasi-Birth-and-Death Process |
| SB | Smart Bernoulli (service policy) |
| $Ser$ | Service Time (random variable) |
| Thr | Threshold (service policy) |
| TPFM | Transition Probability Function Matrix |
| TPM | Transition Probability Matrix |
| UWC | Unobserved Waiting Customer |

# Chapter 1

# Introduction

## 1.1 What's in a Queueing Model?

A queueing model describes the process in which customers (or jobs) contend with each other for a system's resources in the form of one or more servers. Typically, this involves the customers present in the system at a given time waiting in one or more lines, or queues, until they are able to enter service. The design of a queueing model may become very complex depending on the its features, which may or may not be common in other types of models. At the heart of every queueing model, however, is some form of a customer-server relationship, and so it is necessary to be able to describe how they come to find each other, as well as how the server treats the customer (e.g., if there are multiple levels of priorities, or if the server takes vacations) and how long it will take to serve them. In this way we must understand what is meant by an arrival or service process.

The arrival process of a queueing model describes the distribution of the interarrival times of customers, as well as how many customers arrive simultaneously. Arrivals may be individual (e.g., Boxma [18]) or in batches (e.g., Boxma and Groenendijk [20]). The most common assumed arrival process is the *Poisson process*, in which the interarrival times for individual customers of a common type, or class, are independent and identically distributed as exponential random variables with a shared rate. This distributional assumption is often prized for its memoryless property, among other features, which allow for easier analysis through the use of the PASTA property (Poisson Arrivals See Time Averages, Wolff [98]). Specifically, PASTA allows one to equate the steady-state distribution of a model to the distribution of the system at arrival instants, simplifying waiting time analysis.

The service process of a queueing model constitutes the distribution of a customer's service time, as well as the order in which the server(s) attend to customers within their queue, the order in which the server(s) visit different queues if there are multiple queues (i.e., the *polling order*), and how many customers they serve during a visit (i.e., the *service discipline*). In many models, there are no restrictions on what a service time distribution may be, other than it be non-negative with finite moments. In these cases the distribution used in their analysis is referred to as *general.* However, in the scope of this document, to enable us the ability to work within the framework of *matrix analytic methods* (MAM), we elect to use *phase-type* distributions, which shall be introduced in Section 1.2.3.

Some possible choices for the rules governing the service order of customers within the same queue are *first-come, first-served* (FCFS, alternatively denoted as FIFO for *first-in, first-out*), *last-come, first-served* (LCFS), *service in random order* (SIRO), *shortest job first* (SJF), or *most profitable job first*. While a SJF discipline may be optimal from a total system optimization standpoint (Schrage [84]), it requires the assumption that the server knows exactly how long it would take to service each queued customer, which is unrealistic outside of some computer systems models (where the customers are data to be processed and have a known size). It therefore should not be surprising to know that FCFS is the typical standard, as it ensures a level of *fairness* for all customers. The topic of scheduling visits to multiple queues by a server, and the decision of how many customers to serve during a visit, are key features of *polling models*, especially within the analysis of system optimization. For this reason, we choose to table this discussion until Section 1.2.7, where we introduce and discuss polling systems in more detail.

The number of servers, and the number of waiting places within a queue, are also considerations that are addressed in the definition of any queueing model. The choice to have a single server is common, especially in polling models, though a model may in fact assume up to an infinite number of servers (i.e., each customer who arrives immediately begins service). A model with an infinite number of servers may represent, for example, the number of people with the flu (as everyone has their own immune system), or simply a system with a sufficiently large number of servers in relation to the arrival rates of customers (such as tourists asking locals on the street of a large city for directions). A queueing model by default may assume that there are an infinite number of waiting spaces for arriving customers, which is referred to as having an *infinite buffer capacity*. If, on the contrary, there is a cap on the number of customers that may simultaneously wait in the same queue, it is said to have a *finite buffer capacity*. Depending on the model, analysis may require the use of a finite buffer for computational purposes, even if the real life system it is describing does not have such a restriction. In these cases, an infinite buffer system may be approximated through the increase of the number of waiting spaces until the probability of a customer being *blocked* (i.e., arriving to find their queue full and being turned away) is sufficiently small.

Some examples of other features that may be present in a queueing model to more accurately describe a customer's actions are:

- *Balking*: When a customer decides to not join a queue after their arrival if the queue length is too long (e.g., Drekic and Woolford [32]),

- *Jockeying*: When a customer decides to change which queue they are waiting in (e.g., Gertsbakh [37]),

- *Reneging*: When a customer decides to leave a queue before reaching service due to impatience (e.g., Section 1.2.5),

- *Retrials*: When a customer decides to return to a queue after some random delay, having previously left the queue before receiving service (e.g., Artalejo et al. [5]),

- *Routing*: When a customer decides to immediately rejoin a queueing system after completing service (e.g., Towsley [90]).

We would be remiss to not also list some features that can also generalize a server's behaviour within the context of a queueing model, such as:

- *Switchover Times*: When a server is working on a multi-queue system, a switchover time may be incurred between consecutive visits to two different queues (e.g., Servi [85]),

- *Vacations*: When a server takes a break from serving any customers, despite the possible presence of customers waiting in a queue to be served (e.g., Igaki [46]).

Due to the versatility of what we can define as customers, and the services they receive, queueing theory as a whole is adaptable to many real world applications. For instance, aside from the obvious parallels to retail that people experience in their everyday lives, it can be used to model problems in telecommunications (e.g., Palm [73]), traffic (e.g., Boon [15]), health care (e.g., Drekic et al. [31]), production (e.g., Koenigsberg and Mamer [56]), and maintenance (e.g., Mack et al. [66]). Of these, maintenance models are of a particular relevance to the work within Chapters 2 and 3 of this thesis.

## 1.2 Using Matrix Analytic Methods

### 1.2.1 Generating the Generator

The use of MAM combines our knowledge of *continuous-time Markov chains* (CTMCs) or *discrete-time Markov chains* (DTMCs) with our understanding of queueing systems. This thesis focuses solely on modelling systems in continuous time, so we will therefore begin this subsection with a brief review of CTMCs (e.g., Ross [82], Chapter 6).

**Definition:** A stochastic process $\{X(t), t \geq 0\}$ is called a continuous-time Markov chain if the following conditions hold true:

(1) The state space $\mathcal{S}$ of $\{X(t), t \geq 0\}$ is at most countable (i.e., finite $\mathcal{S} = \{0, 1, \ldots, n\}$ or countable $\mathcal{S} = \{0, 1, \ldots\}$). That is, $X(t)$, $t \geq 0$, is a discrete random variable.

(2) (*Markov Property*) For any $s, t \geq 0$, $i, j \in \mathcal{S}$,

$$P(\underbrace{X(t+s)}_{\text{future}} = j|\underbrace{X(s)}_{\text{present}} = i, \underbrace{X(u) = x(u), 0 \leq u < s}_{\text{past}}) = P(X(t+s) = j|X(s) = i),$$

where $x(u) \in \mathcal{S}$ represents the (possibly varying) state of the CTMC in the past as a function of time $u$. That is, the probabilistic properties of the future development of the CTMC only depends on the current state and is independent of its past.

A CTMC is referred to as being *time-homogeneous* if $P(X(t+s) = j|X(s) = i)$ is independent of $s$, in which case we define the *transition probability function*

$$P_{i,j}(t) = P(X(t+s) = j|X(s) = i) = P(X(t) = j|X(0) = i), \ t \geq 0, \ i, j \in \mathcal{S}.$$

We will only be considering CTMCs that are time-homogeneous within this thesis, so this property will be assumed going forward.

The duration of time that a CTMC spends visiting a state $i \in \mathcal{S}$ prior to transitioning to a different state $j \neq i$ is random, having an exponential distribution whose parameter may be state-dependent. We denote this as a *sojourn time* at state $i$, $T_i \sim \text{Exp}(v_i)$, such that $T_i$ is exponentially distributed with rate $v_i$, where $v_i$ is the total of all transition rates leaving state $i$. If $v_i = 0$, then state $i$ is considered to be an *absorbing state* whose sojourn time will be infinite in duration.

Supposing that $v_i > 0$, after sojourn time $T_i$ completes, the CTMC will transition to state $j \neq i$ with probability $p_{i,j}$, $\sum_{j \in \mathcal{S}} p_{i,j} = 1$. If one were to only observe state transitions (but no sojourn times), then the observed movements of this stochastic process can be modelled by its *embedded DTMC* $\{X_n, n \in \mathbb{N}\}$ having *transition probability matrix* (TPM) $P = [p_{i,j}]_{i,j \in \mathcal{S}}$, whose elements are these transition probabilities. A restriction observed in these embedded DTMCs is that $p_{i,i} = 0$ for all $i \in \mathcal{S}$, so long as $v_i > 0$. If $v_i = 0$, then by convention we set $p_{i,i} = 1$ so that state $i$ is absorbing in both the CTMC as well as its embedded DTMC.

Next, define $q_{i,j} = v_i p_{i,j}$, $j \neq i$, as the *probability flow* or *instantaneous rate of transition* from state $i$ to state $j$. It immediately follows that

$$\sum_{\substack{j \in \mathcal{S} \\ j \neq i}} q_{i,j} = v_i \sum_{\substack{j \in \mathcal{S} \\ j \neq i}} p_{i,j} = v_i, \; i \in \mathcal{S}.$$

These names may be understood as follows. Suppose that the initial distribution of a CTMC is $\underline{\alpha}_0 = (\alpha_{0,0}, \alpha_{0,1}, \ldots)$, where $\alpha_{t,i} = P(X(t) = i)$, $i \in \mathcal{S}$, $t \geq 0$. If we let $h > 0$ be a small amount of time such that

$$P(\geq 2 \text{ transitions in } [0,h] | X(0) = i) = o(h), \; \forall \; i \in \mathcal{S},$$

where $o(h)$ denotes a function where

$$\lim_{h \to 0} \frac{o(h)}{h} = 0,$$

then applying Taylor series expansion,

$$\begin{aligned}
&P(1 \text{ transition in } [0,h] | X(0) = i) \\
&= 1 - P(0 \text{ transitions in } [0,h] | X(0) = i) - P(\geq 2 \text{ transitions in } [0,h] | X(0) = i) \\
&= 1 - P(T_i > h) - o(h) \\
&= 1 - e^{-v_i h} + o(h) \\
&= 1 - \left( \sum_{n=0}^{\infty} \frac{(-v_i h)^n}{n!} \right) + o(h) \\
&= v_i h + o(h),
\end{aligned}$$

and for $j \neq i$,

$$\begin{aligned}
P_{i,j}(h) &= P(X(h) = j | X(0) = i, 0 \text{ transitions in } [0,h]) P(0 \text{ transitions in } [0,h] | X(0) = i) \\
&\quad + P(X(h) = j | X(0) = i, 1 \text{ transition in } [0,h]) P(1 \text{ transition in } [0,h] | X(0) = i) \\
&\quad + P(X(h) = j | X(0) = i, \geq 2 \text{ transitions in } [0,h]) P(\geq 2 \text{ transitions in } [0,h] | X(0) = i) \\
&= 0 + p_{i,j}(v_i h + o(h)) + o(h) \\
&= q_{i,j} h + o(h). \qquad (1.1)
\end{aligned}$$

Similarly, we can show that

$$P_{i,i}(h) = P(T_i > h) + o(h) = e^{-v_i h} + o(h) = 1 - v_i h + o(h). \qquad (1.2)$$

It now follows that

$$\begin{aligned}
P(X(h)=j) &= \sum_{i\in\mathcal{S}} P(X(h)=j|X(0)=i)P(X(0)=i) \\
&= \sum_{i\in\mathcal{S}} \alpha_{0,i}P_{i,j}(h) \\
&= \sum_{\substack{i\in\mathcal{S}\\ i\neq j}} \alpha_{0,i}(q_{i,j}h+o(h)) + \alpha_{0,j}(1-v_jh+o(h)) \\
&= \alpha_{0,j} - \alpha_{0,j}v_jh + \sum_{\substack{i\in\mathcal{S}\\ i\neq j}} \alpha_{0,i}q_{i,j}h + o(h). \qquad (1.3)
\end{aligned}$$

That is, over a small time interval of length $h$, we (approximately) observe probability mass moving (or *flowing*) from every state $i \neq j$ into state $j$ at a proportional rate equal to $q_{i,j}$, while probability mass is leaving state $j$ (and flowing to other states) at a proportional rate equal to $v_j = \sum_{k\neq j} q_{j,k}$. For the second name, following steps similar to what we used to obtain Equation (1.1), it is straightforward to confirm that for $j \neq i$,

$$\text{E}[\text{Transitions into } j \text{ in } [0,h]|X(0)=i] = q_{i,j}h + o(h),$$

and so the (expected) rate of transitions into state $j$, given that $X(0)=i$, as $h\to 0$ is

$$\lim_{h\to 0} \frac{\text{E}[\text{Transitions into } j \text{ in } [0,h]|X(0)=i]}{h} = \lim_{h\to 0} \frac{q_{i,j}h+o(h)}{h} = q_{i,j}.$$

Therefore, $q_{i,j}$ can be considered as the instantaneous rate of transition into $j$ from $i$.

Let us now consider how to solve for the *transition probability function matrix* (TPFM) $P(t) = [P_{i,j}(t)]_{i,j\in\mathcal{S}}$, $t \geq 0$. First, note that

$$P_{i,j}(0) = P(X(0)=j|X(0)=i) = \begin{cases} 0 & , \text{ if } i \neq j, \\ 1 & , \text{ if } i = j, \end{cases}$$

implying that $P(0) = I$, where $I$ is the identity matrix. From Equations (1.1) and (1.2), we have

$$\lim_{h\to 0} \frac{P_{i,j}(h)}{h} = \lim_{h\to 0} \frac{q_{i,j}h+o(h)}{h} = q_{i,j},$$

and

$$\lim_{h\to 0} \frac{P_{i,i}(h)-1}{h} = \lim_{h\to 0} \frac{-v_ih+o(h)}{h} = -v_i.$$

Applying the *Chapman-Kolmogorov equations* for CTMCs,

$$P_{i,j}(s+t) = \sum_{k\in\mathcal{S}} P_{i,k}(s)P_{k,j}(t),$$

or in matrix form, $P(s+t) = P(s)P(t)$, we can obtain the *Kolmogorov Backward equations* (KBE)

$$P'(t) = \lim_{h\to 0} \frac{P(t+h)-P(t)}{h} = \lim_{h\to 0} \frac{(P(h)-P(0))P(t)}{h} P(t) = Q \cdot P(t) \qquad (1.4)$$

and *Kolmogorov Forward equations* (KFE)

$$P'(t) = \lim_{h\to 0} \frac{P(t+h) - P(t)}{h} = \lim_{h\to 0} \frac{P(t)(P(h) - P(0))}{h} = P(t) \cdot Q, \tag{1.5}$$

where the matrix $Q$ is referred to as the *generator* (or *infinitesimal generator*) of $\{X(t), t \geq 0\}$, defined as follows.

**Definition:** If $\{X(t), t \geq 0\}$ is a CTMC with TPFM $P(t)$, matrix $Q$ is the infinitesimal generator matrix of $\{X(t), t \geq 0\}$ if

$$Q = \lim_{h\to 0} \frac{P(h) - P(0)}{h} = \begin{array}{c|cccc} & 0 & 1 & 2 & \cdots \\ 0 & -v_0 & q_{0,1} & q_{0,2} & \cdots \\ 1 & q_{1,0} & -v_1 & q_{1,2} & \cdots \\ 2 & q_{2,0} & q_{2,1} & -v_2 & \ddots \\ \vdots & \vdots & \vdots & \ddots & \ddots \end{array} . \tag{1.6}$$

This construction implies that from just knowledge of the generator matrix $Q$, we know that a sojourn time at state $i$ is exponentially distributed with rate $-Q_{i,i} = v_i$, and the transition probability from state $i$ to state $j$ is $-Q_{i,j}/Q_{i,i} = q_{i,j}/v_i = p_{i,j}$. We may also note that all row sums of $Q$ are zero, since

$$\sum_{j\neq i} q_{i,j} = \sum_{j\neq i} v_i p_{i,j} = v_i \sum_{j\neq i} p_{i,j} = v_i \sum_{j\in\mathcal{S}} p_{i,j} = v_i.$$

**Remark 1.1.** The derivation of the KBE and KFE require the interchanging of a limit and a matrix product. This is always justifiable for both equations in the case of a finite state space $\mathcal{S}$, or in the KBE when the state space is countable. Details on how to derive the entry-wise form of the KBE in the countable state space case are provided in the Appendix.

Equations (1.4) and (1.5) provide differential equations which may be solved to obtain the TPFM. We can confirm that $P(t) = e^{tQ}$ satisfies both, where '$e$' represents the *matrix exponential function*, defined as

$$e^{tQ} = I + tQ + \frac{t^2Q^2}{2} + \cdots = \sum_{n=0}^{\infty} \frac{t^n}{n!} Q^n,$$

which reduces to the standard Taylor series expansion of an exponential function when $Q$ is scalar. The marginal distribution of $\{X(t), t \geq 0\}$ can now be evaluated as $\underline{\alpha}_t = \underline{\alpha}_0 P(t)$, $t \geq 0$.

Rather than the marginal distribution of a CTMC, a primary interest within this work is to obtain the *stationary* (or *steady-state*) distribution of a CTMC, and show the connection between it and the generator $Q$.

**Definition:** If $\{X(t), t \geq 0\}$ is a CTMC with TPFM $P(t)$, a probability distribution $\{\pi_i\}_{i\in\mathcal{S}}$, $\pi_i \geq 0$, $\forall\ i \in \mathcal{S}$, is called a stationary distribution of $\{X(t), t \geq 0\}$ if probability row vector $\underline{\pi} = (\pi_0, \pi_1, \ldots)$ satisfies:

(1) (*Normalization Condition*) $\underline{\pi}\,\underline{e}' = \sum_{i\in\mathcal{S}} \pi_i = 1$,

(2) (*Stationary Condition*) $\underline{\pi} = \underline{\pi}P(t)$, $\forall\ t \geq 0$,

where $\underline{e}'$ is a column vector of ones of appropriate length.

Note that in general, the notation $'$ will be used to denote matrix (or vector) transpose, and so $\underline{e}$ represents a row vector of ones. This vector of probabilities is called 'stationary' because if we let $\underline{\alpha}_0 = \underline{\pi}$, then

$$\underline{\alpha}_t = \underline{\alpha}_0 P(t) = \underline{\pi}P(t) = \underline{\pi}.$$

Additionally, this implies that if at any point in time the marginal distribution of $\{X(t), t \geq 0\}$ is equal to probability vector $\underline{\pi}$, then the CTMC has stabilized, and the marginal distribution will remain the same forever, so that $P(X(t) = i) = \pi_i$.

Recalling the definition of $Q$, we can see that if $\mathcal{S}$ has finite-many states, then

$$\begin{aligned}
\underline{\pi} &= \underline{\pi}P(h), \\
\underline{0} &= \underline{\pi}(P(h) - I), \\
\underline{0} &= \lim_{h\to 0} \underline{\pi}_i \frac{P(h) - P(0)}{h} = \underline{\pi}Q, \qquad (1.7)
\end{aligned}$$

where $\underline{0}$ is a row vector of zeroes of appropriate length. This provides us with a much easier way to solve for $\underline{\pi}$, as we do not need to first solve for its TPFM, making the application of Equation (1.7) the preferred method for solving for the stationary distribution. For the countable state-space case, if we assume that $\underline{\pi}$ satisfies $\underline{\pi}Q = \underline{0}$ and let $\underline{\alpha}_0 = \underline{\pi}$, then applying Equation (1.4) we observe that

$$\frac{d}{dt}\underline{\alpha}_t = \frac{d}{dt}\underline{\alpha}_0 P(t) = \underline{\alpha}_0 P'(t) = \underline{\alpha}_0 QP(t) = \underline{\pi}QP(t) = \underline{0}P(t) = \underline{0}.$$

That is, due to the KBE, the marginal distribution of $\{X(t), t \geq 0\}$ does not change in time, implying that $\underline{\pi}$ is a stationary distribution.

Before ending this subsection, it would benefit us to consider yet another interpretation for the probability flows $q_{i,j}$. The following is perhaps more useful when actually constructing the generator for a given model.

**Corollary 1.1.** For a CTMC $\{X(t), t \geq 0\}$ with infinitesimal generator matrix $Q$, we may consider each $q_{i,j}$ as the rate of an independent exponential timer (such that if $q_{i,j} = 0$, then the timer takes on a value of infinity with probability 1). When the CTMC begins a sojourn time at a non-absorbing state $i \in \mathcal{S}$, an independent $\text{Exp}(q_{i,j})$ timer is started for all $j \in \mathcal{S}$. After observing the first timer completion, the CTMC ends its visit to state $i$ and transitions to the shortest timer's respective state $j \neq i$, after which this process repeats itself. Therefore, so long as at least one $q_{i,j} > 0$, we can effectively ignore timers with $q_{i,j} = 0$, as they will never have the shortest duration. If $q_{i,j} = 0$ for all $j \in \mathcal{S}$ (i.e., $v_i = 0$), then the visit to state $i$ never ends as no timer ever finishes, in which case state $i$ is an absorbing state.

This alternative interpretation of the probability flows $q_{i,j}$ result in the exact same probabilistic behaviour for $\{X(t), t \geq 0\}$ as outlined above, which follows due to the unique characteristics of independent exponential distributions. The case of an absorbing state with $v_i = 0$ is clearly in agreement with our understood behaviour of an absorbing state, so let us investigate the cases of states $i \in \mathcal{S}$ that are non-absorbing. We can confirm this by checking: (1) the distribution of a sojourn time in state $i$; (2) the transition probabilities out of state $i$; (3) the state selection of the transition process out of state $i$ is independent of the length of the sojourn time in state $i$. The proofs of these three claims are presented in the Appendix. Given our new understanding of model parameters $q_{i,j}$, we are ready for our first example.

### 1.2.2 Example 1: Analyzing the $M/M/1$ Queueing Model, a Birth-and-Death Process

Our first illustration on how to apply MAM within a queueing theory context will examine how to construct the infinitesimal generator $Q$ for an $M/M/1$ queue, as well as how to solve for the steady-state distribution for this given $Q$ using Equation (1.7). The name '$M/M/1$' informs us of the queueing model's main characteristics, and is an example of Kendall's notation [51]. In general, we may label a queueing system as '$A/S/m/c/p$', which tells us the interarrival distribution ($A$), service distribution ($S$), number of servers ($m$), total number of waiting and service spaces for customers ($c$), and the size of the customer population ($p$). If $c$ or $p$ are omitted, as in the case of this $M/M/1$ model, we assume that their values are infinity. Here, the letter '$M$' stands for *memoryless* (or *Markovian*), and indicates that both the interarrival and service times follow exponential distributions. We assume by default that the system is under a FCFS discipline and that the random service and interarrival times are independent.

We denote the rates of the exponentially distributed interarrival and service times by $\lambda$ and $\mu$, respectively. They are both assumed to be positive constants that do not vary with the length of the queue. We let the CTMC $\{X(t), t \geq 0\}$ track the number of customers in the system at time $t$, such that its state space is $\mathcal{S} = \mathbb{N}$. In order to construct the infinitesimal generator matrix $Q$ for this CTMC, as per Equation (1.6), we need to know the values of $v_i$ and $q_{i,j}$ for $i, j \in \mathbb{N}$.

Regardless of queue length, the Poisson process arrival flow of customers always acts upon the system. Meanwhile, the lone server will always tend to a customer so long as the queue is not empty, immediately beginning the next service after a service time completion if another waiting customer is available. As customers arrive and receive service individually, we may only observe the state of the system increase by 1 or decrease by 1 in a single transition. This makes the $M/M/1$ queue an example of a *birth-and-death process*, with births corresponding to arrivals and deaths corresponding to departures from a non-empty queue after service completions. This implies that $q_{i,j} = 0$, $\forall\ |i - j| > 1$.

When the queue is empty (i.e., $X(t) = 0$), the only distribution actively acting upon the system is the Exp($\lambda$) distributed interarrival time of the next customer. After observing the next arrival (i.e., after this exponential timer completes), the state of the system changes to 1. By Corollary 1.1, this implies that $q_{0,1} = \lambda$. Since $q_{0,j} = 0$, $\forall\ j > 1$, it holds that

$$v_0 = \sum_{\substack{j \in \mathbb{N} \\ j \neq 0}} q_{0,j} = q_{0,1} = \lambda.$$

That is, the sojourn time spent in state 0 is simply equal in distribution to an interarrival time.

When $X(t) = i \in \mathbb{Z}^+$, there are active exponential timers for both interarrival and service times whose completions would change the queue length, and hence, the state of the system. If an arrival is observed first, the state of the system will increment to $i+1$, so we have $q_{i,i+1} = \lambda$. If a service completion is observed first, the state of the system will decrement to $i - 1$, so we have $q_{i,i-1} = \mu$. It immediately follows that

$$v_i = \sum_{\substack{j \in \mathbb{N} \\ j \neq i}} q_{i,j} = q_{i,i-1} + q_{i,i+1} = \mu + \lambda, \ i \in \mathbb{Z}^+,$$

reflecting the fact that the sojourn time spent in state $i$ is the minimum of these two timers. For this specific model, after observing an exponential timer complete, the slower timer probabilistically restarts by the *memoryless property* of exponential distributions, while the faster timer is replaced by an iid timer, with the exception of a service completion that empties the queue.

We can now construct $Q$ as

$$Q = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \end{array} \!\!\begin{array}{c} \begin{array}{cccc} 0 & 1 & 2 & \cdots \end{array} \\ \left[\begin{array}{cccc} -\lambda & \lambda & 0 & \cdots \\ \mu & -(\lambda+\mu) & \lambda & \cdots \\ 0 & \mu & -(\lambda+\mu) & \ddots \\ \vdots & \vdots & \ddots & \ddots \end{array}\right] \end{array}.$$

Applying Equation (1.7), it immediately follows that

$$\begin{aligned} 0 &= -\lambda\pi_0 + \mu\pi_1, \\ 0 &= \lambda\pi_{i-1} - (\lambda+\mu)\pi_i + \mu\pi_{i+1}, \ i \in \mathbb{Z}^+. \end{aligned}$$

Note that we can re-express these equations in terms of the probability flow leaving and entering a particular state, obtaining

$$\begin{aligned} \lambda\pi_0 &= \mu\pi_1, \\ (\lambda+\mu)\pi_i &= \lambda\pi_{i-1} + \mu\pi_{i+1}, \ i \in \mathbb{Z}^+. \end{aligned}$$

The interpretation for these expressions is as follows. In order for the distribution of $X(t)$ to be stationary, the probability mass at each state must remain constant in time. We have previously observed in Equation (1.3) how the probability mass in a given state changes in a small interval of time $h$ due to the flow of probability leaving and entering that state. If we suppose that $\underline{\alpha}_0 = \underline{\pi}_0$, then Equation (1.3) becomes

$$\begin{aligned} P(X(h) = j) &= \pi_j - \pi_j v_j h + \sum_{\substack{i \in \mathcal{S} \\ i \neq j}} \pi_i q_{i,j} h + o(h) \\ &= \begin{cases} \pi_0 - (\lambda\pi_0 - \mu\pi_1)h + o(h) & \text{, if } j = 0, \\ \pi_j - ((\lambda+\mu)\pi_j - \lambda\pi_{j-1} - \mu\pi_{j+1})h + o(h) & \text{, if } j \in \mathbb{Z}^+. \end{cases} \end{aligned}$$

If the balance equations did not hold true, then this would indicate that the probability mass in a given state $j \in \mathcal{S}$ is not stable over small time intervals, and hence $\underline{\pi}$ would not be stationary.

In order to solve this system of equations, it is easy to show that $\pi_i = \frac{\lambda}{\mu}\pi_{i-1}, i \in \mathbb{Z}^+$. Defining $\rho = \lambda/\mu$ as the *traffic intensity* (or *workload*) of the queue, we find

$$\pi_i = \rho\pi_{i-1} = \rho^2\pi_{i-2} = \cdots = \rho^i\pi_0, i \in \mathbb{Z}^+.$$

Applying the normalization condition ($\underline{\pi}\,\underline{e}' = 1$), so long as $\rho < 1$, we have

$$1 = \sum_{i=0}^{\infty} \pi_i = \sum_{i=0}^{\infty} \rho^i \pi_0 = \pi_0 \sum_{i=0}^{\infty} \rho^i = \frac{\pi_0}{1-\rho},$$

implying that $\pi_0 = 1 - \rho$, and in general,

$$\pi_i = (1-\rho)\rho^i, \; i \in \mathbb{N}.$$

We recognize this as a geometric distribution with a probability of success equal to $1 - \rho$.

Note that the steady-state distribution will exist iff $\rho = \lambda/\mu < 1$. From the theory of Poisson processes (e.g., Ross [82], Theorem 5.1) , we know that the expected number of customer arrivals in a single time unit equals $\lambda$. As the expected quantity of work that a single customer requires is equal to $1/\mu$, we can interpret $\rho$ as the expected amount of work for the server (in time units) that arrives every time unit. If $\rho < 1$, then the system is stable since the server will periodically be able to empty the queue of all customers (and we may prove that the CTMC is *positive recurrent*). If $\rho > 1$, then over time the server will surely fall behind on their work, and the queue length will grow to infinity (implying that there will be a final visit time for every state and the system is *transient*). In the event that $\rho = 1$, there is no expected drift in queue length over time (and we may prove that the CTMC is *null recurrent*).

The parameter $\rho$ may also be understood as the *server utilization* of the queue. That is, $\rho$ is the long-run proportion of time that the server is busy. To see why, we require the theory of alternating renewal processes (e.g., Ross [82], Section 7.5.1). Consider a system which can be described as being in one of two states at a particular moment, say on or off, and the time it spends in a state is an iid state-dependent random variable, $Y_n \sim Y$ (for the $n^{\text{th}}$ on time) or $Z_n \sim Z$ (for the $n^{\text{th}}$ off time). If we suppose that it is turned on at time zero, then the system will probabilistically restart itself at every future instance of switching from off to on, and in the long run it will hold that

$$P(\text{System is on}) = \frac{\mathrm{E}[Y]}{\mathrm{E}[Y] + \mathrm{E}[Z]}.$$

In the context of an $M/M/1$ queue, we let the 'on' state represent the server working and the 'off' state represent the server being idle. Now, it is clear that the server will only be idle from the time that the queue empties until the next arrival, and so $Z \sim \mathrm{Exp}(\lambda)$, with $\mathrm{E}[Z] = \lambda^{-1}$. The time from an arrival until a queue is emptied again is referred to as a *busy period*, which is a random variable we shall denote by $BP$. In order to find the expected value of $BP$, we will derive its *Laplace-Stieltjes transform* (LST).

The LST for a random variable $X$ is simply defined as $\tilde{F}_X(s) = \mathrm{E}[e^{-sX}]$. One benefit of deriving the LST is that it can be used to obtain the moments of $X$. Observe that if we differentiate the LST $r$ times with respect to $s$, we obtain

$$\begin{aligned}\frac{d^r}{ds^r}\tilde{F}_X(s) &= \frac{d^r}{ds^r}\mathrm{E}[e^{-sX}]\\ &= \frac{d^r}{ds^r}\mathrm{E}\left[\sum_{n=0}^{\infty}\frac{(-s)^n}{n!}X^n\right]\\ &= \frac{d^r}{ds^r}\sum_{n=0}^{\infty}\frac{(-s)^n}{n!}\mathrm{E}[X^n]\\ &= (-1)^r\sum_{n=r}^{\infty}\frac{n\cdot(n-1)\cdots(n-r+1)}{n!}(-s)^{n-r}\mathrm{E}[X^n].\end{aligned}$$

If we let $s \to 0$, all but the first remaining term of the above sum vanishes, leaving $(-1)^r\mathrm{E}[X^r]$. Thus, it follows that we may obtain the $r^{\text{th}}$ moment from the formula

$$\mathrm{E}[X^r] = (-1)^r\frac{d^r}{ds^r}\,\tilde{F}_X(s)\Big|_{s=0},\ r \in \mathbb{Z}^+. \tag{1.8}$$

Considering now the structure of a busy period, it will be comprised of the service time of the first arriving customer, as well as all busy periods started by customers who arrive during this service (which may be none). Letting $Ser$ denote the random service time of the customer who arrived to find the queue empty, $N$ denote the random number of customers who arrived during $Ser$ (such that $N|(Ser = t) \sim \mathrm{Poi}(\lambda t)$, i.e., a Poisson distribution with mean $\lambda t$, by the theory of Poisson processes), and $BP_n \sim BP$ be the (iid) random busy period started by the $n^{\text{th}}$ customer to arrive during $Ser$, we have

$$\tilde{F}_{BP}(s) = \mathrm{E}[e^{-sBP}] = \mathrm{E}[e^{-s(Ser+\sum_{n=1}^{N}BP_n)}],$$

where (by convention) we let $\sum_{n=1}^{0}BP_n = 0$. First, note that

$$\begin{aligned}\mathrm{E}[\mathrm{E}[e^{-s\sum_{n=1}^{N}BP_n}|N, Ser]|Ser] &= \mathrm{E}[\tilde{F}_{BP}(s)^N|Ser]\\ &= \sum_{m=0}^{\infty}\tilde{F}_{BP}(s)^m\cdot\frac{(\lambda Ser)^m}{m!}e^{-\lambda Ser}\\ &= e^{-\lambda Ser}e^{\lambda Ser\tilde{F}_{BP}(s)}.\end{aligned}$$

Applying the law of total expectation, we obtain

$$\tilde{F}_{BP}(s) = \mathrm{E}[\mathrm{E}[e^{-s\cdot Ser}e^{-s\sum_{n=1}^{N}BP_n}|Ser]] = \mathrm{E}[e^{-(s+\lambda-\lambda\tilde{F}_{BP}(s))Ser}] = \tilde{F}_{Ser}(s+\lambda-\lambda\tilde{F}_{BP}(s)).$$

Now, by Equation (1.8) and the fact that for any random variable $X$, $\tilde{F}_X(0) = \mathrm{E}[e^0] = 1$, it follows that

$$\begin{aligned}\mathrm{E}[BP] &= -\frac{d}{ds}\,\tilde{F}_{BP}(s)\Big|_{s=0}\\ &= -\frac{d}{ds}\,\tilde{F}_{Ser}(s+\lambda-\lambda\tilde{F}_{BP}(s))\Big|_{s=0}\\ &= -\tilde{F}'_{Ser}(s+\lambda-\lambda\tilde{F}_{BP}(s))[1-\lambda\tilde{F}'_{BP}(s)]\Big|_{s=0}\\ &= -\tilde{F}'_{Ser}(0)[1-\lambda\tilde{F}'_{BP}(0)]\\ &= \mathrm{E}[Ser](1+\lambda\mathrm{E}[BP]).\end{aligned}$$

Rearranging for $\mathrm{E}[BP]$, we get

$$\mathrm{E}[BP] = \frac{\mathrm{E}[Ser]}{1 - \lambda \mathrm{E}[Ser]},$$

which for the $M/M/1$ queue results in

$$\mathrm{E}[BP] = \frac{1/\mu}{1 - \lambda/\mu} = \frac{1}{\mu - \lambda}, \tag{1.9}$$

provided that $\mu > \lambda$.

Returning to our alternating renewal process, we had $\mathrm{E}[Z] = \lambda^{-1}$, and now we know that $\mathrm{E}[Y] = \mathrm{E}[BP] = (\mu - \lambda)^{-1}$. Thus, in the long run, we have

$$P(\text{Server is busy}) = \frac{(\mu - \lambda)^{-1}}{(\mu - \lambda)^{-1} + \lambda^{-1}} = \frac{1}{1 + \frac{\mu - \lambda}{\lambda}} = \frac{1}{\mu/\lambda} = \rho,$$

and

$$P(\text{Server is idle}) = 1 - P(\text{Server is busy}) = 1 - \rho = \pi_0,$$

as required.

**Remark 1.2.** Given the stationary distribution of a queue, we can work backwards to obtain the expected busy period from the idle probability, $\pi_0$. That is, since

$$\pi_0 = \frac{\lambda^{-1}}{\mathrm{E}[BP] + \lambda^{-1}},$$

it follows that

$$\mathrm{E}[BP] = \frac{1 - \pi_0}{\lambda \pi_0}. \tag{1.10}$$

This can be very useful when analyzing a queue which has an analytic solution for its steady-state distribution, but whose busy period is difficult to analyze directly.

### 1.2.3 Continuous Phase-Type Distributions

Before continuing on to our second example, we introduce the continuous phase-type distribution, which plays an important role in many models which use MAM for their analysis. Phase-type distributions were introduced by Neuts in 1975 [69] as a generalization of the exponential distribution. In this sub-subsection we will define what constitutes a continuous phase-type distribution and list some of their key properties. For an in-depth look at phase-type distributions, see for example He, Chapter 1 [43].

A continuous phase-type distribution is defined as the time until absorption in a CTMC having at least one absorbing state. If a CTMC has multiple absorbing states, for the purposes of the time until absorption, they may be combined and treated as a single state without affecting this time (as we do not care what state the CTMC is absorbed into, just *when*). Therefore, for the purposes of this sub-subsection, let us assume that a CTMC has exactly one absorbing state (labelled as state 0), and $M$ transient states $1, 2, \ldots, M$, such that we may express its infinitesimal generator matrix by

$$Q = \begin{array}{c|c|cccc} & 0 & 1 & 2 & \cdots & M \\ 0 & 0 & 0 & 0 & \cdots & 0 \\ \hline 1 & q_{1,0} & -v_1 & q_{1,2} & \cdots & q_{1,M} \\ 2 & q_{2,0} & q_{2,1} & -v_2 & \ddots & q_{2,M} \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ M & q_{M,0} & q_{M,1} & q_{M,2} & \cdots & -v_M \end{array} = \left[\begin{array}{c|c} 0 & \underline{0} \\ \hline \underline{S}_0' & S \end{array}\right],$$

where $S$ is a $M \times M$ square matrix corresponding to the transient portion of $Q$ and $\underline{S}_0' = -S\underline{e}'$ is the column vector of absorption rates having length $M$. We alternatively refer to $S$ as a *subgenerator* (or *rate matrix*), as it is a component of $Q$ and is not an infinitesimal generator itself (note that at least one row of $S$ must not have a row sum of zero).

Applying block-wise matrix multiplication, we find that

$$Q^n = \left[\begin{array}{c|c} 0 & \underline{0} \\ \hline -S^n\underline{e}' & S^n \end{array}\right], \; n \in \mathbb{Z}^+.$$

This convenient form allows us to easily calculate the TPFM of this CTMC. Recalling that

$$P(t) = e^{tQ} = \sum_{n=0}^{\infty} \frac{t^n}{n!} Q^n, \; t \geq 0,$$

where $Q^0 = I$, it follows that

$$P(t) = I + \sum_{n=1}^{\infty} \frac{t^n}{n!} \left[\begin{array}{c|c} 0 & \underline{0} \\ \hline -S^n\underline{e}' & S^n \end{array}\right] = \left[\begin{array}{c|c} 1 & \underline{0} \\ \hline (I - e^{St})\underline{e}' & e^{St} \end{array}\right], \; t \geq 0.$$

Let the initial probability vector of the CTMC be

$$\underline{\alpha}_0 = (\alpha_{0,0}, \underbrace{\alpha_{0,1}, \alpha_{0,2}, \ldots, \alpha_{0,M}}_{\underline{\alpha}_0^*}),$$

where $\alpha_{0,0}$ is the probability of beginning in the absorbing state and $\underline{\alpha}_0^*$ is the transient portion of $\underline{\alpha}_0$. The probability that the CTMC has not reached its absorption state by some $t \geq 0$ is simply the probability of it being in any one of its $M$ transient states at that time. Thus, letting $T$ denote the continuous phase-type random variable, since $\underline{\alpha}_t = \underline{\alpha}_0 P(t)$, it follows that

$$P(T > t) = P(X(t) \in \{1, 2, \ldots, M\}) = \underline{\alpha}_0^* e^{St} \underline{e}', \; t \geq 0.$$

Thus, the cumulative distribution function (CDF) of $T$ is

$$F_T(t) = 1 - \underline{\alpha}_0^* e^{St} \underline{e}', \; t \geq 0, \tag{1.11}$$

and its probability density function (PDF) on the positive real number line is

$$f_T(t) = \frac{d}{dt} F_T(t) = \frac{d}{dt}(1 - \underline{\alpha}_0^* e^{St} \underline{e}') = -\underline{\alpha}_0^* \left[\frac{d}{dt} e^{St}\right] \underline{e}' = -\underline{\alpha}_0^* e^{St} S \underline{e}' = \underline{\alpha}_0^* e^{St} \underline{S}_0', \; t > 0, \tag{1.12}$$

where we may interchange the order of vector-matrix product and derivative since we are only considering a finite state space. In summary, we define these distributions as follows:

**Definition:** The time until absorption in a CTMC, $T$, having probability mass at zero $P(T = 0) = 1 - \underline{\alpha}_0^* \underline{e}'$ and PDF

$$f_T(t) = \underline{\alpha}_0^* e^{St} \underline{S}_0', \ t > 0,$$

is said to follow what is called a continuous phase-type distribution of order $M$, which is typically denoted by

$$T \sim \mathrm{PH}_M(\underline{\alpha}_0^*, S),$$

where $M$ is the number of transient states (i.e., the dimension of $S$, the transient part of the infinitesimal generator matrix), and $\underline{\alpha}_0^*$ is the part of the initial distribution corresponding to transient states.

From the above definition, we may find the LST and moments of a continuous phase-type distribution. However, before deriving the LST of a phase-type distribution, we require the following properties of matrix exponential functions. First, recall that for scalar exponential functions, it holds that $e^a e^b = e^{a+b}$. If we now suppose that $A$ and $B$ are square matrices of equal dimension satisfying $AB = BA$, then we have (applying the binomial expansion)

$$e^{A+B} = \sum_{n=0}^{\infty} \frac{1}{n!}(A+B)^n = \sum_{n=0}^{\infty} \frac{1}{n!} \sum_{k=0}^{n} \binom{n}{k} A^k B^{n-k} = \sum_{n=0}^{\infty} \sum_{k=0}^{n} \frac{1}{k!} A^k \cdot \frac{1}{(n-k)!} B^{n-k}.$$

Switching the order of summation and letting $m = n - k$, we find

$$e^{A+B} = \sum_{k=0}^{\infty} \sum_{n=k}^{\infty} \frac{1}{k!} A^k \cdot \frac{1}{(n-k)!} B^{n-k} = \left( \sum_{k=0}^{\infty} \frac{1}{k!} A^k \right) \left( \sum_{m=0}^{\infty} \frac{1}{m!} B^m \right) = \left( \sum_{k=0}^{\infty} \frac{1}{k!} A^k \right) e^B = e^A e^B.$$

Thus, under certain restrictions, we have $e^A e^B = e^B e^A = e^{A+B} = e^{B+A}$ for the matrix exponential function. Next, consider the product of scalar exponential $e^a$ with matrix exponential $e^B$. Applying the Taylor expansion of the scalar exponential function along with the fact that $I^n = I$ for an identity matrix $I$,

$$e^a e^B = \left( \sum_{n=0}^{\infty} \frac{a^n}{n!} \right) e^B = \left( \sum_{n=0}^{\infty} \frac{a^n}{n!} \right) I e^B = \left( \sum_{n=0}^{\infty} \frac{a^n}{n!} I^n \right) e^B = e^{aI} e^B = e^{aI+B}.$$

Finally, assume that $S$ is a subgenerator from an absorbing CTMC. It follows that $S$ is an invertible square matrix (as not all row sums of $S$ equal zero, 0 is not an eigenvalue of $S$, and hence $S$ is invertible), the anti-derivative of $e^{St}$ is

$$\int e^{St} dt = \int \sum_{n=0}^{\infty} \frac{t^n}{n!} S^n dt = \sum_{n=0}^{\infty} \frac{t^{n+1}}{(n+1)!} S^n (S S^{-1}) = \left( \sum_{m=0}^{\infty} \frac{t^m}{m!} S^m - I \right) S^{-1} = \left( e^{St} - I \right) S^{-1}.$$

Since $e^{St}$ contains the conditional probabilities of being in transient states at time $t$, $\lim_{t \to \infty} e^{St} = \mathbf{0}$, as the CTMC will eventually end up in its absorbing state with probability 1. Thus,

$$\int_0^{\infty} e^{St} dt = \lim_{t \to \infty} \left( e^{St} - I \right) S^{-1} - \left( e^{\mathbf{0}} - I \right) S^{-1} = -S^{-1}.$$

We may now derive the LST of $T$ directly. As $T$ has a mixed distribution, it follows that

$$\begin{aligned}\tilde{F}_T(s) = \mathrm{E}[e^{-sT}] &= e^{-s\cdot 0}P(T=0) + \int_0^\infty e^{-st} f_T(t)dt \\ &= 1 \cdot \alpha_{0,0} + \int_0^\infty e^{-st}\underline{\alpha}_0^* e^{St}\underline{S}_0' dt \\ &= 1 - \underline{\alpha}_0^*\underline{e}' + \underline{\alpha}_0^*\left(\int_0^\infty e^{(S-sI)t}dt\right)\underline{S}_0' \\ &= 1 - \underline{\alpha}_0^*\underline{e}' - \underline{\alpha}_0^*(S-sI)^{-1}\underline{S}_0' \\ &= 1 - \underline{\alpha}_0^*\underline{e}' + \underline{\alpha}_0^*(sI-S)^{-1}\underline{S}_0',\ s \geq 0, \end{aligned} \tag{1.13}$$

where we recognize that if $s \geq 0$, then $S - sI$ is a valid subgenerator for a continuous phase-type distribution.

To obtain the general formula for the moments of $T$, we must take derivatives of Equation (1.13) with respect to $s$. Note that

$$\begin{aligned}\frac{d}{ds}(sI-S)^{-1} &= -\frac{d}{ds}(I-sS^{-1})^{-1}S^{-1} \\ &= -\frac{d}{ds}\left(\sum_{n=0}^\infty s^n S^{-n}\right)S^{-1} \\ &= -\left(\sum_{n=1}^\infty n s^{n-1}S^{-n}\right)S^{-1} \\ &= -\left(\sum_{n=0}^\infty (n+1)s^n S^{-(n+1)}\right)S^{-1} \\ &= -\left(S^{-1}\sum_{n=0}^\infty s^n S^{-n} + sS^{-1}\sum_{n=1}^\infty n s^{n-1}S^{-n}\right)S^{-1} \\ &= -(I-sS^{-1})^{-1}S^{-2} + sS^{-1}\frac{d}{ds}(sI-S)^{-1} \\ &= -(I-sS^{-1})^{-2}S^{-2} \\ &= -(sI-S)^{-2}. \end{aligned}$$

Applying the product rule, we can prove by induction that

$$\frac{d}{ds}(sI-S)^{-k} = -k(sI-S)^{-(k+1)},\ k \in \mathbb{Z}^+.$$

Thus, for $r \in \mathbb{Z}^+$,

$$\begin{aligned}
\frac{d^r}{ds^r}\tilde{F}_T(s) &= \frac{d^r}{ds^r}\left(1 - \underline{\alpha}_0^*\underline{e}' + \underline{\alpha}_0^*(sI - S)^{-1}\underline{S}_0'\right) \\
&= \frac{d^r}{ds^r}\underline{\alpha}_0^*(sI - S)^{-1}\underline{S}_0' \\
&= \frac{d^{r-1}}{ds^{r-1}}(-1)\underline{\alpha}_0^*(sI - S)^{-2}\underline{S}_0' \\
&\vdots \\
&= \frac{d}{ds}(-1)^{r-1}(r-1)!\underline{\alpha}_0^*(sI - S)^{-r}\underline{S}_0' \\
&= (-1)^r r!\underline{\alpha}_0^*(sI - S)^{-(r+1)}\underline{S}_0',
\end{aligned}$$

and so the $r^{\text{th}}$ moment of $T$ is

$$\mathrm{E}[T^r] = (-1)^r \frac{d^r}{ds^r}\left.\tilde{F}_T(s)\right|_{s=0} = r!\underline{\alpha}_0^*(-S)^{-(r+1)}\underline{S}_0' = (-1)^r r!\underline{\alpha}_0^* S^{-r}\underline{e}', \; r \in \mathbb{Z}^+, \tag{1.14}$$

where we applied the equation $\underline{S}_0' = -S\underline{e}'$ in the last equality.

In some queueing models, a distribution (typically service times) may be left unspecified, with just the assumptions that its first two moments are finite and that it is non-negative. This is then referred to as a general distribution, denoted by a '$G$' in Kendall's notation. The incentive is that one can later select any distribution of interest and make use of the results without changing any of the analysis. Unfortunately, we are not able to build a general distribution into a generator to use with MAM, however we do have the following result from Asmussen ([6], Proposition 2):

**Proposition:** The class $\mathscr{PH}$ of phase-type distributions is dense (in the sense of weak convergence) in the class $\mathscr{P}$ of all distributions on $(0, \infty)$.

This proposition implies that in theory, continuous phase-type distributions can be used to approximate any non-negative distribution within any desired accuracy. For information on fitting phase-type distributions to a given distribution or to observed data, see Asmussen et al. [7]. Correspondingly, we use '$PH$' in Kendall's notation for systems assuming phase-type distributed interarrival or service times. While in practice there are computational limitations restricting the order (i.e., number of phases) of a phase-type distribution that may be used in a queueing model, which can reduce how closely you can approximate certain distributions, this does not negate the value of the theory. With continual advancements in both computer computation speed and memory, these restrictions loosen over time.

We close this sub-subsection by listing the general representations of some common phase-type distributions which will appear later in this work.

- (Exp) Exponential distribution: $f(x; \lambda) = \lambda e^{-\lambda x}, x > 0, \lambda > 0$,

$$X \sim \mathrm{PH}_1\left(\underline{\alpha}_0^* = 1, \; S = -\lambda\right).$$

- (E$_k$) Erlang-$k$ distribution: $f(x;k,\lambda) = \frac{\lambda^k x^{k-1} e^{-\lambda x}}{(k-1)!}, x > 0, \lambda > 0, k \in \mathbb{Z}^+$,

$$X \sim \mathrm{PH}_k \left( \underline{\alpha}_0^* = (1, \underline{0})\,,\ S = \begin{array}{c|cccccc|} & 1 & 2 & 3 & \cdots & k-1 & k \\ 1 & -\lambda & \lambda & 0 & \cdots & 0 & 0 \\ 2 & 0 & -\lambda & \lambda & \cdots & 0 & 0 \\ 3 & 0 & 0 & -\lambda & \ddots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots & \vdots \\ k-1 & 0 & 0 & 0 & \cdots & -\lambda & \lambda \\ k & 0 & 0 & 0 & \cdots & 0 & -\lambda \end{array} \right),$$

  where $\underline{0}$ has a length of $k-1$.

- (H$_k$) Hyperexponential-$k$ distribution: $f(x;k,\lambda_1,\ldots,\lambda_k) = \sum_{i=1}^k \lambda_i e^{-\lambda_i x} p_i, x > 0, \lambda_i > 0, k \in \mathbb{Z}^+$ where $p_i$ are probabilities satisfying $\sum_{i=1}^k p_i = 1, p_i \geq 0$,

$$X \sim \mathrm{PH}_k \left( \underline{\alpha}_0^* = (p_1, p_2, \ldots, p_k)\,,\ S = \begin{array}{c|cccc|} & 1 & 2 & \cdots & k \\ 1 & -\lambda_1 & 0 & \cdots & 0 \\ 2 & 0 & -\lambda_2 & \ddots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ k & 0 & 0 & \cdots & -\lambda_k \end{array} \right).$$

**Remark 1.3.** We may alternatively derive the LST of $T$ through first step analysis, rather than applying properties of matrix exponential functions. Let

$$\tilde{F}_{T,i}(s) = \mathrm{E}[e^{-sT} | X(0) = i],\ i = 0, 1, \ldots, M,$$

implying that

$$\tilde{F}_T(s) = \sum_{i=0}^{M} \alpha_{0,i} \tilde{F}_{T,i}(s),$$

due to the law of total expectation. It is clear that $\tilde{F}_{T,0}(s) = e^{-s(0)} = 1$, so let us consider the case of an initial state $i$ which is transient. After sojourn time $T_i$, the CTMC will either transition to the absorbing state or to a different transient state. We may decompose the conditional distribution of the time until absorption $T$ into the sum of $T_i$ and $T_i^*$, where $T_i^*$ denotes the (independent) remaining time until absorption after the sojourn in state $i$, having the following mixture distribution:

$$T_i^* \sim \begin{cases} 0 & \text{, with probability } [\underline{S}_0']_i / v_i, \\ T|(X(0) = j) & \text{, with probability } q_{i,j}/v_i,\ j \neq i, \end{cases}$$

where $[\underline{S}_0']_i$ is the absorption rate out of state $i$ (i.e., the $i^{\text{th}}$ element of $\underline{S}_0'$). Thus, by independence of $T_i$ and $T_i^*$,

$$\tilde{F}_{T,i}(s) = \mathrm{E}[e^{-sT} | X(0) = i] = \mathrm{E}[e^{-s(T_i + T_i^*)}] = \mathrm{E}[e^{-sT_i}]\mathrm{E}[e^{-sT_i^*}]. \tag{1.15}$$

Since $T_i \sim \mathrm{Exp}(v_i)$, we have for $s > -v_i$,

$$\mathrm{E}[e^{-sT_i}] = \int_0^\infty e^{-st} v_i e^{-v_i t} dt = \frac{v_i}{v_i + s} \int_0^\infty (v_i + s) e^{-(v_i+s)t} dt = \frac{v_i}{v_i + s}, \tag{1.16}$$

while

$$\mathrm{E}[e^{-sT_i^*}] = \frac{[\underline{S}_0']_i}{v_i} e^{-s(0)} + \sum_{\substack{j=1 \\ j \neq i}}^{M} \frac{q_{i,j}}{v_i} \tilde{F}_{T,j}(s). \tag{1.17}$$

Substituting Equations (1.16) and (1.17) into Equation (1.15), we obtain

$$(v_i + s)\tilde{F}_{T,i}(s) = [\underline{S}_0']_i + \sum_{\substack{j=1 \\ j \neq i}}^{M} q_{i,j} \tilde{F}_{T,j}(s).$$

Moving all conditional LST terms to the left side, we have

$$s\tilde{F}_{T,i}(s) - \left( -v_i \tilde{F}_{T,i}(s) + \sum_{\substack{j=1 \\ j \neq i}}^{M} q_{i,j} \tilde{F}_{T,j}(s) \right) = [\underline{S}_0']_i, \;\; i = 1, 2, \ldots, M,$$

which in matrix form is

$$(sI - S) \left[ \begin{array}{cccc} \tilde{F}_{T,1}(s) & \tilde{F}_{T,2}(s) & \cdots & \tilde{F}_{T,M}(s) \end{array} \right]' = \underline{S}_0'.$$

Thus, the (unconditional) LST of $T \sim \mathrm{PH}_M(\underline{\alpha}_0^*, S)$ is

$$\tilde{F}_T(s) = \alpha_{0,0}\tilde{F}_{T,0}(s) + \underline{\alpha}_0^* \left[ \begin{array}{cccc} \tilde{F}_{T,1}(s) & \tilde{F}_{T,2}(s) & \cdots & \tilde{F}_{T,M}(s) \end{array} \right] = 1 - \underline{\alpha}_0^* \underline{e}' + \underline{\alpha}_0^* (sI - S)^{-1} \underline{S}_0',$$

as required.

### 1.2.4 Example 2: Analyzing the $M/PH/1$ Queueing Model, a Level-Independent QBD

Generalizing on our previous example in Section 1.2.2 concerning the $M/M/1$ queueing model, we replace our assumption of exponentially distributed service times with the assumption of service times that are iid $\mathrm{PH}_k(\underline{\alpha}_0^*, S)$ random variables with the restriction that $\underline{\alpha}_0^* \underline{e}' = 1$ (i.e., they are strictly positive in duration). Here, we elect to denote the order by '$k$' rather than '$M$' to avoid confusion with the notation for our interarrival process. We shall henceforth refer to this as the $M/PH/1$ queueing model. As an exponential distribution is an example of a continuous phase-type of order 1, we did not require the tracking of a service phase. However, as we now allow phase-types of any order, we must track the current service phase in addition to the queue length. Accurately tracking the service phase is important, as the remaining time until the active service completes will depend on the current phase. For example, the residual service time of a customer whose service requirement follows an Erlang-2 distribution will of course follow an Erlang-2 distribution if they are observed in their first service phase, but the residual time will simply follow an exponential distribution if they are observed in their second phase.

Therefore, since knowledge of service phases impact future developments of the system, we must track the current service phase to maintain the Markov property. In addition to letting $X(t) \in \mathbb{N}$ represent the number of customers in the system at time $t$, we now let $Y(t) \in \{1, 2, \dots, k\}$ indicate the current phase of the service distribution at time $t$. It is straightforward to confirm that the pair $\{(X(t), Y(t)), t \geq 0\}$ is a CTMC.

From its definition, a phase-type distribution involves a specific CTMC with subgenerator $S$. It follows that the times between state changes (i.e., phase changes) in this CTMC follow independent exponential distributions with rates equal to the negatives of the main-diagonal elements of $S$, while $S$'s off-diagonal elements correspond to rates of competing exponential timers. This allows $S$ (along with $\underline{S}_0'$) to be inserted as elements into an infinitesimal generator matrix.

When modelling this type of system, we consider a generator matrix as a collection of *blocks*, or *submatrices*. Keeping $\lambda$ as the rate of the exponential interarrival process and letting $\mathbf{0}$ represent an appropriately dimensioned matrix of zeroes, the generator of $\{(X(t), Y(t)), t \geq 0\}$ is

$$Q = \begin{array}{c|cccccc|} & 0 & 1 & 2 & 3 & 4 & \cdots \\ 0 & -\lambda & \lambda\underline{\alpha}_0^* & \underline{0} & \underline{0} & \underline{0} & \cdots \\ 1 & \underline{S}_0' & S - \lambda I_k & \lambda I_k & \mathbf{0} & \mathbf{0} & \cdots \\ 2 & \underline{0}' & \underline{S}_0'\,\underline{\alpha}_0^* & S - \lambda I_k & \lambda I_k & \mathbf{0} & \cdots \\ 3 & \underline{0}' & \mathbf{0} & \underline{S}_0'\,\underline{\alpha}_0^* & S - \lambda I_k & \lambda I_k & \ddots \\ 4 & \underline{0}' & \mathbf{0} & \mathbf{0} & \underline{S}_0'\,\underline{\alpha}_0^* & S - \lambda I_k & \ddots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \ddots \end{array}. \tag{1.18}$$

Note that the indexing on the rows and columns indicate the value of the queue length, such that the $(i, j)^{\text{th}}$ block of $Q$ (which we may denote by $Q_{i,j}$) contains all transitions where $X(t)$ can change from $i$ to $j$ in one step. As the queue length is the outer-most index, we refer to it as the *level* of the process. Since customers arrive and receive service individually, the level can only change by at most 1 in a given transition, and so $Q_{i,j} = \mathbf{0}$ if $|i - j| > 1$. For this reason, we refer to this type of CTMC as a *quasi-birth-and-death process* (QBD). The state space of $\{(X(t), Y(t)), t \geq 0\}$ is

$$\mathcal{S} = \{(0, 0)\} \cup \{(X, Y) : X \in \mathbb{Z}^+, Y \in \{1, 2, \dots, k\}\},$$

where we let $Y(t)$ take a placeholder value of 0 when there is no customer currently undergoing service (and hence, the service phase has no observable value). Correspondingly, there is only the single sublevel of 0 in level 0, while each positive level contains $k$ ordered sublevels corresponding to the value of $Y(t)$. Therefore, $Q_{0,0}$ is a scalar while $Q_{i,i}$, $i \in \mathbb{Z}^+$, are $k \times k$ square matrices.

When the CTMC is in level 0, there are no customers in the queue, and so the sojourn time in state $(0, 0)$ simply follows an $\text{Exp}(\lambda)$ distribution. It then immediately follows that $Q_{0,0} = -\lambda$. Upon observing a customer arrival to an empty queue, they immediately enter service. However, unlike the $M/M/1$ queue, we must initialize the phase of their service according to initial probability vector $\underline{\alpha}_0^*$. If we apply the *splitting* or *thinning* property of Poisson processes, then we can separate arriving customers according to their initial service phases which are determined in an iid manner. Thus, we can consider competing exponential interarrival times with rates $\lambda\alpha_{0,i}$, $i = 1, 2, \dots, k$. If the first interarrival time to complete corresponded to one with rate $\lambda\alpha_{0,j}$, then its service initializes in phase $j$, and so the CTMC must transition to state $(1, j)$.

We therefore define

$$Q_{0,1} = \lambda \left[ \begin{array}{cccc} \alpha_{0,1} & \alpha_{0,2} & \cdots & \alpha_{0,k} \end{array} \right] = \lambda \underline{\alpha}_0^*.$$

Consider now level 1, and suppose that the CTMC is in state $(1, j)$, $j = 1, 2, \ldots, k$. Arrivals to a non-empty queue will not affect the current service phase of the lead customer. Therefore, if the Exp($\lambda$) timer is next to complete, the CTMC will transition from state $(1, j)$ to $(2, j)$, and so we let $Q_{1,2} = \lambda I_k$, where $I_k$ is an identity matrix of dimension $k \times k$. While in state $(1, j)$, there is an exponential timer with rate $[\underline{S}_0']_j$ whose completion indicates that the phase-type distributed service has completed. As there are no further customers waiting in the queue (whose initial service phase we would need to determine), this results in a transition to state $(0, 0)$, and we must have $Q_{1,0} = \underline{S}_0'$. All remaining possible transitions are for transitions to transient phases within the customer's service time distribution, with the completion of the Exp($q_{j,i}$) timer resulting in a transition from state $(1, j)$ to state $(1, i)$ (these do not change the level of the CTMC). As the sojourn time in state $(1, j)$ is the minimum of all the timers we have considered, it must have an Exp($-S_{j,j} + \lambda$) distribution. Correspondingly, we have $Q_{1,1} = S - \lambda I_k$, where the identity matrix allows us to subtract $\lambda$ from the negative main diagonal elements of $S$ while leaving the $q_{j,i}$ elements unchanged and in the correct positions.

For higher levels of this CTMC, a single customer's arrival or service completion does not result in changing from an empty queue to a non-empty queue (or vice versa), so we have a consistent relationship between the $Q_{i,i-1}$, $Q_{i,i}$, and $Q_{i,i+1}$ blocks, $i = 2, 3, \ldots$ (and hence, this is in fact a *level-independent QBD*). In fact, for the same reasons outlined above, we have $Q_{i,i} = Q_{1,1}$ and $Q_{i,i+1} = Q_{1,2}$, for all $i \in \mathbb{Z}^+$. Therefore, let us consider being in state $(i, j)$ and suppose that the next observed event is a customer departure. With probability $\alpha_{j,l}$, the next customer in line was an arrival from the thinned Poisson process corresponding to customers whose initial service phase is $l$. As their initial service phase is independent of everything else, the joint probability of a transition representing the service completion of the customer out of the $j^{\text{th}}$ service phase and the next customer starting their service in phase $l$ is

$$P_{(i,j),(i-1,l)} = \frac{[\underline{S}_0']_j}{-S_{j,j} + \lambda} \alpha_{0,l},$$

which is the one-step transition probability in the embedded DTMC from state $(i, j)$ to state $(i - 1, l)$. Multiplying $P_{(i,j),(i-1,l)}$ by the exponential rate of the sojourn time distribution in state $(i, j)$ provides us with our required exponential rate, $[\underline{S}_0']_j \alpha_{0,l}$, and so in matrix form we have $Q_{i,i-1} = \underline{S}_0' \underline{\alpha}_0^*$.

Now that we have our infinitesimal generator matrix, our next goal is to find the steady-state distribution of this queueing system. Due to the block nature of the generator, it is convenient to define steady-state probability row vectors $\underline{\pi}_i$ relating to states within level $i$, $i \in \mathbb{N}$. For this particular model, level 0 contains a single state, and hence $\underline{\pi}_0$ is a scalar. Letting $\pi_{i,j}$ denote the steady-state joint probability that the CTMC is in state $(i, j)$, we define

$$\underline{\pi} = (\underline{\pi}_0, \underline{\pi}_1, \underline{\pi}_2, \ldots),$$

where $\underline{\pi}_0 = \pi_{0,0}$ and

$$\underline{\pi}_i = (\pi_{i,1}, \pi_{i,2}, \ldots, \pi_{i,k}), \ i \in \mathbb{Z}^+.$$

Rather than examining the calculations required for this specific $M/PH/1$ model, we will illustrate how to obtain $\underline{\pi}$ from $\underline{\pi} Q = \underline{0}$ (and $\underline{\pi}\, \underline{e}' = 1$) for general level-independent QBDs. We consider a generator of the form

$$Q = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \\ \vdots \end{array}\!\!\begin{array}{c} \begin{array}{cccccc} 0 & 1 & 2 & 3 & 4 & \cdots \end{array} \\ \left[\begin{array}{cccccc} B_{00} & B_{01} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots \\ B_{10} & B_{11} & A_0 & \mathbf{0} & \mathbf{0} & \cdots \\ \mathbf{0} & A_2 & A_1 & A_0 & \mathbf{0} & \ddots \\ \mathbf{0} & \mathbf{0} & A_2 & A_1 & A_0 & \ddots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots \end{array}\right] \end{array}, \tag{1.19}$$

where all blocks $B$ and $A$ are constructed legally such that all main diagonal elements of $Q$ are negative, off-diagonal elements are non-negative, and row sums are zero. Note that there may be more than one type of sublevel within the system, and that the probability row vectors $\underline{\pi}_i$ are constructed in a logical fashion, containing the ordered steady-state probabilities for all states within level $i$. Also, we assume that there are finite-many states within each level. From the relationship $\underline{\pi}Q = \underline{0}$ and the constraint that the probabilities must sum to one, we obtain the series of matrix equations

$$\underline{0} = \underline{\pi}_0 B_{00} + \underline{\pi}_1 B_{10}, \tag{1.20}$$
$$\underline{0} = \underline{\pi}_0 B_{01} + \underline{\pi}_1 B_{11} + \underline{\pi}_2 A_2, \tag{1.21}$$
$$\underline{0} = \underline{\pi}_i A_0 + \underline{\pi}_{i+1} A_1 + \underline{\pi}_{i+2} A_2, \ i \in \mathbb{Z}^+, \tag{1.22}$$
$$1 = \underline{\pi}\,\underline{e}'.$$

In the $M/M/1$ model, we found that the steady-state probabilities expressed a geometric relationship

$$\pi_i = \rho\pi_{i-1} = \cdots = \rho^i\pi_0, \ i \in \mathbb{Z}^+,$$

which followed as a result of consistent tridiagonal elements of $Q$ (for levels 1 and higher). Here, we have a similar looking generator, with the exception that we have blocks rather than scalar elements, and the consistency is assumed for levels 2 and higher. We therefore make the assumption that for some square matrix $R$, we have the *matrix-geometric* relationship

$$\underline{\pi}_i = \underline{\pi}_{i-1} R = \cdots = \underline{\pi}_1 R^{i-1}, \ i \in \mathbb{Z}^+, \tag{1.23}$$

where we let $R^0 = I$. Substituting this assumption into Equation (1.22), we have

$$\begin{aligned} \underline{0} &= \underline{\pi}_i A_0 + \underline{\pi}_{i+1} A_1 + \underline{\pi}_{i+2} A_2 \\ &= \underline{\pi}_1 R^{i-1} A_0 + \underline{\pi}_1 R^i A_1 + \underline{\pi}_1 R^{i+1} A_2 \\ &= \underline{\pi}_1 R^{i-1}(A_0 + RA_1 + R^2 A_2). \end{aligned}$$

For a solution to exist, we cannot have $\underline{\pi}_1 = \underline{0}$ or $R = \mathbf{0}$, so it must hold that

$$A_0 + RA_1 + R^2 A_2 = \mathbf{0}. \tag{1.24}$$

Equation (1.24) is referred to as the *matrix quadratic equation*, and it has been shown that matrix $R$ is the solution which, entry-wise, has the smallest non-negative elements (e.g., Neuts [70], Theorem 1.7.1). In general, there is no analytic closed form solution for $R$ (although it may be found for some special cases), but it may be found iteratively. Rearranging Equation (1.24), if we let

$$R(n) = -A_0 A_1^{-1} - R^2(n-1) A_2 A_1^{-1}, \; n \in \mathbb{Z}^+, \tag{1.25}$$

and $R(0) = \mathbf{0}$, then $\{R(n)\}_{n=1}^{\infty}$ will entry-wise monotonically converge to the true value of $R$, such that the $(i,j)^{\text{th}}$ element of $R(n)$ is less than or equal to the $(i,j)^{\text{th}}$ element of $R$. After calculating $R$, it can be used to solve Equations (1.20) and (1.21) for $\underline{\pi}_0$ and $\underline{\pi}_1$, namely

$$\begin{bmatrix} \underline{\pi}_0 & \underline{\pi}_1 \end{bmatrix} \begin{bmatrix} B_{00} & B_{01} \\ B_{10} & B_{11} + RA_2 \end{bmatrix} = \underline{0}. \tag{1.26}$$

The normalization condition may be represented by

$$\underline{\pi}\,\underline{e}' = \underline{\pi}_0 \underline{e}' + \sum_{i=1}^{\infty} \underline{\pi}_i\, \underline{e}' = \underline{\pi}_0 \underline{e}' + \underline{\pi}_1 \sum_{i=1}^{\infty} R^{i-1} \underline{e}' = \underline{\pi}_0 \underline{e}' + \underline{\pi}_1 (I - R)^{-1} \underline{e}'. \tag{1.27}$$

By combining Equations (1.26) and (1.27), we obtain a system with one more linear equation than unknown,

$$\begin{bmatrix} \underline{\pi}_0 & \underline{\pi}_1 \end{bmatrix} \begin{bmatrix} B_{00} & B_{01} & \underline{e}' \\ B_{10} & B_{11} + RA_2 & (I-R)^{-1}\underline{e}' \end{bmatrix} = \begin{bmatrix} \underline{0} & 1 \end{bmatrix}. \tag{1.28}$$

Hence, if we arbitrarily drop a column (other than the right-most column corresponding to the normalization condition) such that the resulting square matrix has an inverse, as well as one of the zeroes from the right-hand vector, the values of $\underline{\pi}_0$ and $\underline{\pi}_1$ may be found by post-multiplying both sides by said inverse. The remaining $\underline{\pi}_i$'s may then be obtained via Equation (1.23).

While we now know how to solve for the steady-state distribution of a level-independent QBD, it is important to consider when such a solution exists. For the $M/M/1$ queue, we simply require the traffic intensity $\rho = \lambda/\mu$ to be less than 1. However, we no longer have simple scalars to perfectly describe both the arrival and service processes in a $M/PH/1$ queue. If we consider the level-independent QBD process at moderate-to-high queue lengths (if the queue is spending any long-run fraction of time in boundary states, it follows that it must be stable), then we can describe the state transitions between its sublevels using the modified infinitesimal generator matrix $A = A_0 + A_1 + A_2$. It should be easy to see that generator $A$ maintains the standard properties of an infinitesimal generator matrix, such that it has row sums of zero, negative main diagonal elements, and non-negative off-diagonal elements.

If we find the steady-state distribution of a CTMC with generator $A$, then it will tell us the fraction of time that our level-independent QBD spends in each of its sublevels (when far from the boundary level 0, e.g., if the queue length were to go to infinity). Define $\underline{\nu}$ as the steady-state probability row vector satisfying

$$\begin{aligned} \underline{0} &= \underline{\nu} A, \\ 1 &= \underline{\nu}\, \underline{e}'. \end{aligned} \tag{1.29}$$

After solving for $\underline{\nu}$, it may be used in conjunction with $A_0$ and $A_2$ to determine the mean drifts to higher or lower levels of the CTMC when far from level 0. A necessary and sufficient condition for stability is for the drift to lower levels to be greater than that to higher levels, a parallel to the condition $\mu > \lambda$ in the $M/M/1$ model. The drifts are defined as

$$\text{Drift Up} = \underline{\nu} A_0 \underline{e}',$$

and

$$\text{Drift Down} = \underline{\nu} A_2 \underline{e}',$$

so we can determine that the system is stable, and hence $\underline{\pi}$ will exist, iff

$$\underline{\nu} A_2 \underline{e}' > \underline{\nu} A_0 \underline{e}'.$$

We conclude this example by examining the stability condition of the $M/PH/1$ queueing model. The service times are distributed as $\mathrm{PH}_k(\underline{\alpha}_0^*, S)$ and by Equation (1.14) have an expected value of $-\underline{\alpha}_0^* S^{-1} \underline{e}'$. Let $\mu$ be the rate of service completions (i.e., the inverse of the mean service time), so that

$$\mu = \frac{-1}{\underline{\alpha}_0^* S^{-1} \underline{e}'}.$$

Comparing the generators from Equations (1.18) and (1.19), it is clear that:

$$\begin{aligned} B_{00} &= -\lambda, \ B_{01} = \lambda \underline{\alpha}_0^*, \\ B_{10} &= \underline{S}_0', \ \ B_{11} = S - \lambda I_k, \ A_0 = \lambda I_k, \\ &\qquad\quad A_2 = \underline{S}_0' \underline{\alpha}_0^*, \qquad A_1 = S - \lambda I_k. \end{aligned}$$

Therefore, with

$$A = A_0 + A_1 + A_2 = \lambda I_k + (S - \lambda I_k) + \underline{S}_0' \, \underline{\alpha} = S + \underline{S}_0' \underline{\alpha}_0^*,$$

the first component of Equation (1.29) becomes

$$\underline{0} = \underline{\nu} S + \underline{\nu} \, \underline{S}_0' \, \underline{\alpha}_0^*.$$

Multiplying both sides from the right by $S^{-1}\underline{e}'$, we obtain

$$0 = \underline{\nu} \, \underline{e}' + \underline{\nu} \, \underline{S}_0' \left( \underline{\alpha}_0^* S^{-1} \underline{e}' \right) = 1 - (\underline{\nu} \, \underline{S}_0')/\mu,$$

implying that $\underline{\nu} \, \underline{S}_0' = \mu$. Finally, the drift up is

$$\underline{\nu} A_0 \underline{e}' = \underline{\nu} \lambda I_k \underline{e}' = \lambda \underline{\nu} \, \underline{e}' = \lambda,$$

while the drift down is

$$\underline{\nu} A_2 \underline{e}' = (\underline{\nu} \, \underline{S}_0')(\underline{\alpha}_0^* \underline{e}') = \mu,$$

implying that a necessary and sufficient condition for stability is $\mu > \lambda$, agreeing with the stability condition for the $M/M/1$ queue.

Considering the $M/M/1$ queue, if we let the service times be $\mathrm{PH}_1(\underline{\alpha}_0^* = 1, S = -\mu)$ (i.e., $\mathrm{Exp}(\mu)$), then

$$\begin{aligned} B_{00} &= -\lambda, \ B_{01} = \lambda, \\ B_{10} &= \mu, \quad B_{11} = -(\lambda + \mu), \ A_0 = \lambda, \\ &\qquad\quad A_2 = \mu, \qquad\quad A_1 = -(\lambda + \mu), \end{aligned}$$

and so Equation (1.25) becomes

$$R(n) = -A_0 A_1^{-1} - R^2(n-1) A_2 A_1^{-1} = \frac{\lambda}{\lambda + \mu} + \frac{\mu}{\lambda + \mu} R^2(n-1), \ n \in \mathbb{Z}^+,$$

so in the $n^{\text{th}}$ iteration, our approximation of $R$ changes by

$$\begin{aligned}R(n) - R(n-1) &= \frac{\lambda}{\lambda+\mu} + \frac{\mu}{\lambda+\mu}R^2(n-1) - R(n-1)\\ &= \frac{1}{\lambda+\mu}(1 - R(n-1))(\lambda - \mu R(n-1)),\ n \in \mathbb{Z}^+.\end{aligned}$$

Thus, our value of $R(n)$ (which is a scalar) increases with $n$, so long as $R(n-1) < \min\{1, \lambda/\mu\} = \lambda/\mu$, if we assume the stability condition holds. If we suppose that $R(k) = \lambda/\mu - \varepsilon$, $k \in \mathbb{N}$, $\varepsilon > 0$, then

$$\begin{aligned}R(k+1) &= \frac{\lambda}{\lambda+\mu} + \frac{\mu}{\lambda+\mu}\left(\frac{\lambda}{\mu} - \varepsilon\right)^2\\ &= \frac{\lambda}{\lambda+\mu} + \frac{\mu}{\lambda+\mu}\left(\frac{\lambda^2}{\mu^2} - 2\varepsilon\frac{\lambda}{\mu} + \varepsilon^2\right)\\ &= \frac{\lambda}{\mu} - \frac{\mu}{\lambda+\mu}\varepsilon\left(2\frac{\lambda}{\mu} - \varepsilon\right) < \frac{\lambda}{\mu},\end{aligned}$$

so long as $\varepsilon < 2\lambda/\mu$. Since $R(0) = 0$ and $R(n) - R(n-1) > 0$ for $R(n-1) < \lambda/\mu$, the maximum deviation from $\lambda/\mu$ that we can observe is $\varepsilon = \lambda/\mu < 2\lambda/\mu$ at $n = 0$. Therefore,

$$0 = R(0) < R(1) < R(2) < \cdots < \frac{\lambda}{\mu},$$

and so $R(n)$ monotonically converges to

$$\lim_{n\to\infty} R(n) = \frac{\lambda}{\mu} = R,$$

without every exceeding the limiting value, agreeing with our earlier cited theory. Note that we can alternatively solve for $R = \lambda/\mu$ by replacing $R(n)$ and $R(n-1)$ by $R$ in Equation (1.25), and rejecting $R = 1$ as a possible solution as it would imply that $\pi_i = \pi_1$, $i \in \mathbb{Z}^+$.

Now that we have found $R$, Equation (1.28) becomes

$$\begin{aligned}\begin{bmatrix} 0 & 0 & 1 \end{bmatrix} &= \begin{bmatrix} \pi_0 & \pi_1 \end{bmatrix}\begin{bmatrix} -\lambda & \lambda & 1 \\ \mu & \lambda - \mu + \frac{\lambda}{\mu}\mu & \left(1 - \frac{\lambda}{\mu}\right)^{-1} \end{bmatrix}\\ &= \begin{bmatrix} \pi_0 & \pi_1 \end{bmatrix}\begin{bmatrix} -\lambda & \lambda & 1 \\ \mu & -\mu & \frac{\mu}{\mu-\lambda} \end{bmatrix}.\end{aligned}$$

Removing the redundant center column (and its corresponding zero from the left hand side vector), it follows that

$$\begin{bmatrix} \pi_0 & \pi_1 \end{bmatrix} = \begin{bmatrix} 0 & 1 \end{bmatrix}\begin{bmatrix} -\lambda & 1 \\ \mu & \frac{\mu}{\mu-\lambda} \end{bmatrix}^{-1} = \begin{bmatrix} 0 & 1 \end{bmatrix}\begin{bmatrix} -\frac{1}{\mu} & \frac{\mu-\lambda}{\mu^2} \\ \frac{\mu-\lambda}{\mu} & \frac{\lambda(\mu-\lambda)}{\mu^2} \end{bmatrix} = \begin{bmatrix} 1 - \frac{\lambda}{\mu} & \left(1 - \frac{\lambda}{\mu}\right)\left(\frac{\lambda}{\mu}\right) \end{bmatrix},$$

and by Equation (1.23),

$$\pi_i = \pi_1 R^{i-1} = \left(1 - \frac{\lambda}{\mu}\right)\left(\frac{\lambda}{\mu}\right)\left(\frac{\lambda}{\mu}\right)^{i-1} = \left(1 - \frac{\lambda}{\mu}\right)\left(\frac{\lambda}{\mu}\right)^{i},\ i \in \mathbb{Z}^+,$$

recovering the solution from Section 1.2.2.

### 1.2.5 Customer Behaviour: Reneging

In the real world, many factors may cause a customer to prematurely leave their queue prior to receiving service. If the customers in question are perishable goods waiting to be processed, then it would be reasonable to expect that after some amount of time, they would be at risk of spoiling. If the customers are people, then the overwhelming cause of a departure prior to receiving service is caused by impatience. We refer to the event of a customer departure due to their impatience as *reneging*.

Palm [73] was the first to investigate the idea of an impatience function, $I(t)$, for customers waiting in a telecommunications queue. They posited the form of the impatience function for customers waiting in a system without back signaling (i.e., a customer must wait on the line until congestion was reduced, resulting in an increase of their rate of impatience over time), with back signaling (i.e., a customer may hang up after receiving a busy signal and will be called when the lines become free, resulting in a constant rate of impatience), or with busy signals (i.e., upon receiving a busy signal, they must hang up and try again later, resulting in a constant rate of impatience with additional impatience experienced every time they had to redial). In the case of a system without back signaling, they gave evidence to support a quadratic relationship between $I(t)$ and $t$.

In practice, many models simply assume an exponential distribution for their customers' impatience times, treated as a random exponential clock that begins ticking the moment they enter a queue. If this time runs out before the customer reaches a server, they leave their queue and are lost to the system (permanently, or perhaps temporarily in the case of a retrial system), but the customer is assumed safe once they have reached service and are no longer a threat to leave.

Shin and Choo [86] considered such a system with customers, while waiting in the queue to be serviced, are subject to exponentially distributed impatience times. In particular, they considered an $M/M/s$ queue with customers who are subject to not only reneging, but also balking and retrials. Whenever a customer would enter the queue, either as an external arrival or as a retrial from the orbit, they either enter the queue, enter the orbit, or leave the system entirely, with probabilities that can depend on if there are any free servers or if all servers are busy and the number of waiting customers are below or over a given threshold. Due to the possibility of reneging customers entering the orbit, they possess a chance of re-entering the queue at a later time. In order to solve for the steady-state distribution, strategic truncation and approximation was necessary in the form of limiting the size of the orbit and assuming that the total effective reneging rate does not change after a certain queue length.

Drekic et al. [31] also assumed exponential impatience times when studying transplant waiting lists. In their work which was analyzed using matrix analytic methods, the queue is divided into two priority classes dependent on the health of the customer, where the most at-risk customers were placed into the higher priority queue and would receive service (i.e., an organ) before any low priority customers. The customer/patient interarrival times and service times (i.e., organ interarrival times) were assumed to follow independent exponential distributions, with class-dependent parameters. Also, customers in either queue were subject to exponential reneging clocks, where low priority customers who renege would either enter the high priority queue (i.e., self-promote) or leave the system all together, while a reneging customer from the high priority queue was assumed to always be lost. The steady-state probabilities of this system were found, and different types of waiting times were considered. Model parameter fitting to real world data was also addressed.

Altman and Yechiali [4] and Yechiali [100] considered exponential impatience times as well, within the context of models where the servers may be absent due to a vacation period initiated by an emptied queue, or a disaster which simultaneously emptied the queue and required a repair time to be completed before the servers could tend to customers again, respectively. In both models, customers who arrived to find no working servers were at risk of reneging, but only up to the time at which the servers were again operational.

Within Chapters 4 - 6 of this thesis, we consider models where a given customer's exponential reneging rate is allowed to depend on their class as well as their position within their queue. The flexibility of MAM allow us to take into consideration the fact that a target customer's reneging rate may change over the duration of their time spent waiting in their queue, which also impacts their probability of reneging as well as their actual waiting time distribution.

Boxma and De Waal [22] and Sakuma and Takine [83] break from this, allowing their customers' impatience times to follow non-exponential distributions. Boxma and De Waal's model assumes generally distributed reneging times, Poisson process arrivals, generally distributed service times, and $m$ servers. Their main statistic of interest was their model's overflow probability (i.e., the probability of a customer leaving due to impatience). Exact distributional results were impossible given the general reneging distribution, but several methods of how to approximate this probability were discussed. Sakuma and Takine considered a $k$-class system, with class-dependent arrival rates, phase-type service time distributions, and *deterministic* impatience times. A *virtual waiting time* approach was used (i.e., the total workload in a queue caused by customers who will actually reach the server). The stationary distribution of the virtual waiting time was found (using a *level crossing* argument), as well as each classes' loss probabilities, actual waiting time distributions (which we consider for one of our models in Section 4.4), and mean queue lengths.

While the vast majority of work which allows reneging assumes that the abandonment time instants of each customer are independent, it is possible still to apply the concept in systems where epochs of abandonment are in fact not independent. For example, Adan et al. [2] considered a system with Poisson process arrivals, and a server who conducts generally distributed service times, and who leaves for a generally distributed vacation time when the queue empties. Within this model, customers who arrive during the vacation were assumed to observe a unique abandonment epoch at the moment of vacation completion, where every customer simultaneously (and independently) decides if they will leave the queue with the same probability, or were assumed to observe multiple abandonment epochs which occur according to a Poisson process during the vacation period. Due to the simultaneous decisions all using the same constant reneging probability, the number of customers immediately after an abandonment epoch was able to be described using a binomial distribution, depending on the total number of customers in the queue immediately prior to the epoch.

### 1.2.6 Example 3: Analyzing the $M/PH/1+M$ Queueing Model, a Level-Dependent QBD

When constructing the infinitesimal generator matrix of a QBD, it is possible that it will not exhibit the level-independent form observed in Section 1.2.4. That is, there may not exist a particular threshold after which transitions within or out of a given level at or beyond that threshold become independent of the value of that level. For example, if the customers in a queue are at risk of abandonment through reneging, then the total rate of customer reneging will be proportional to the queue length (e.g., if every customer has an independent exponential impatience timer). Another type of system that could result in level dependency is a closed queueing system, as examined in Chapters 2 and 3 of this thesis, where the arrival rate of customers corresponds to the total failure rate of all working machines, which is itself inversely proportional to the queue length of machines waiting to be repaired.

As an illustration of a *level-dependent QBD*, we consider a $M/PH/1+M$ queue, where the '$+M$' notation (e.g., Boxma and De Waal [22]) indicates that customers waiting in the queue (who are not currently being served) have iid exponentially distributed impatience times. We let the rate of these iid impatience times be $\gamma$, and we otherwise keep the same assumptions and notations as the $M/PH/1$ queue from Section 1.2.4. Again letting $X(t) \in \mathbb{Z}^+$ denote the number of customers in the queue and $Y(t) \in \{1, 2, \ldots, k\}$ represent the current phase of service (defined to be zero if there is no customer in service), the pair $\{(X(t), Y(t)), t \geq 0\}$ is again a CTMC with state space

$$\mathcal{S} = \{(0,0)\} \cup \{(X,Y) : X \in \mathbb{Z}^+, Y \in \{1, 2, \ldots, k\}\}.$$

In contrast to the $M/PH/1$ queue, we must now consider an additional competing exponential timer at every state within levels $i = 2, 3, \ldots$ for each waiting customer. As we are simply tracking the total number of customers in the system, given that a customer reneges and leaves the queue, we are indifferent to which waiting customer it was. Thus, we consider the time until the next observed customer departure due to reneging (i.e., the minimum of the active impatience times), which will have an $\text{Exp}((i-1)\gamma)$ distribution. As the departure of a waiting customer has no impact on the active service time's distribution, this would result in a transition from state $(i, j)$ to $(i-1, j)$, $i = 2, 3, \ldots$, $j = 1, 2, \ldots, k$. We incorporate this feature by adding a $(i-1)\gamma I_k$ term to the relevant $Q_{i,i-1}$ blocks, while simultaneously subtracting $(i-1)\gamma$ from the main diagonals. Updating Equation (1.18), the infinitesimal generator matrix for the $M/PH/1+M$ queue is

$$Q = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ \vdots \end{array} \begin{array}{c} \begin{array}{cccccc} 0 & 1 & 2 & 3 & 4 & \cdots \end{array} \\ \begin{bmatrix} -\lambda & \lambda\underline{\alpha}_0^* & \underline{0} & \underline{0} & \underline{0} & \cdots \\ \underline{S}_0' & S-\lambda I_k & \lambda I_k & \mathbf{0} & \mathbf{0} & \cdots \\ \underline{0}' & \underline{S}_0'\,\underline{\alpha}_0^*+\gamma I_k & S-(\lambda+\gamma)I_k & \lambda I_k & \mathbf{0} & \cdots \\ \underline{0}' & \mathbf{0} & \underline{S}_0'\,\underline{\alpha}_0^*+2\gamma I_k & S-(\lambda+2\gamma)I_k & \lambda I_k & \ddots \\ \underline{0}' & \mathbf{0} & \mathbf{0} & \underline{S}_0'\,\underline{\alpha}_0^*+3\gamma I_k & S-(\lambda+3\gamma)I_k & \ddots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots \end{bmatrix} \end{array}. \tag{1.30}$$

In this model, we see that the total rate of transitions to lower levels increase with the level of the system (i.e., the length of the queue), while the rate of transitions to higher levels due to arrivals is constant.

Our goal now becomes to calculate the steady-state distribution of this queueing system,

$$\underline{\pi} = (\underline{\pi}_0, \underline{\pi}_1, \underline{\pi}_2, \ldots),$$

where we let $\underline{\pi}_i$ contain the ordered steady-state probabilities for states within level $i$ (such that $\pi_{i,j}$ is the stationary probability that the CTMC is in state $(i,j)$), such that $\underline{\pi}_0 = \pi_{0,0}$ and

$$\underline{\pi}_i = (\pi_{i,1}, \pi_{i,2}, \ldots, \pi_{i,k}), \ i \in \mathbb{Z}^+.$$

Rather than focusing solely on the $M/PH/1+M$ queue, we will instead illustrate how to solve for these probabilities for the case of a general level-dependent QBD (based on the procedure proposed by Gaver et al. [36]).

Suppose that the infinitesimal generator matrix for a CTMC takes the form

$$Q = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \\ \vdots \end{array} \!\!\begin{array}{c} \begin{array}{cccccc} 0 & 1 & 2 & 3 & 4 & \cdots \end{array} \\ \begin{bmatrix} Q_{0,0} & Q_{0,1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots \\ Q_{1,0} & Q_{1,1} & Q_{1,2} & \mathbf{0} & \mathbf{0} & \cdots \\ \mathbf{0} & Q_{2,1} & Q_{2,2} & Q_{2,3} & \mathbf{0} & \ddots \\ \mathbf{0} & \mathbf{0} & Q_{3,2} & Q_{3,3} & Q_{3,4} & \ddots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots \end{bmatrix} \end{array}, \tag{1.31}$$

such that block $Q_{i,j}$ contain rates corresponding to transitions from states in level $i$ to states in level $j$. From Equation (1.31) and $\underline{\pi}Q = \underline{0}$, we obtain the system of matrix equations

$$\underline{0} = \underline{\pi}_0 Q_{0,0} + \underline{\pi}_1 Q_{1,0}, \tag{1.32}$$

$$\underline{0} = \underline{\pi}_i Q_{i,i+1} + \underline{\pi}_{i+1} Q_{i+1,i+1} + \underline{\pi}_{i+2} Q_{i+2,i+1}, \ i \in \mathbb{N}. \tag{1.33}$$

Generalizing Equation (1.23), we now assume that

$$\underline{\pi}_i = \underline{\pi}_0 \prod_{j=1}^{i} R_j, \ i \in \mathbb{Z}^+. \tag{1.34}$$

Substitution into Equation (1.33) results in

$$\begin{aligned} \underline{0} &= \underline{\pi}_i Q_{i,i+1} + \underline{\pi}_{i+1} Q_{i+1,i+1} + \underline{\pi}_{i+2} Q_{i+2,i+1} \\ &= \underline{\pi}_0 \prod_{j=1}^{i} R_j Q_{i,i+1} + \underline{\pi}_0 \prod_{j=1}^{i+1} R_j Q_{i+1,i+1} + \underline{\pi}_0 \prod_{j=1}^{i+2} R_j Q_{i+2,i+1} \\ &= \underline{\pi}_0 \prod_{j=1}^{i} R_j \left( Q_{i,i+1} + R_{i+1} Q_{i+1,i+1} + R_{i+1} R_{i+2} Q_{i+2,i+1} \right), \ i \in \mathbb{N}. \end{aligned}$$

For a solution to exist, we cannot have $\underline{\pi}_0 = \underline{0}$ or $\prod_{j=1}^{i} R_j = \mathbf{0}$, so (shifting indices) it must hold that

$$Q_{i-1,i} + R_i Q_{i,i} + R_i R_{i+1} Q_{i+1,i} = \mathbf{0}, \ i \in \mathbb{Z}^+.$$

Solving for $R_i$, we have

$$R_i = -Q_{i-1,i} \left( Q_{i,i} + R_{i+1} Q_{i+1,i} \right)^{-1}, \ i \in \mathbb{Z}^+. \tag{1.35}$$

As each $R_i$ references the value of $R_{i+1}$, in order to actually calculate the steady-state probabilities, we must implement a state truncation at some level $b$ by setting $R_i = \mathbf{0}$ for all $j > b$. This is, of course, under the assumption that this does not occur naturally within the infinitesimal generator. If this was true, then no modifications are required. In either case, we will now have $\underline{\pi}_i = \underline{0}$, $i = b+1, b+2, \ldots$, so we redefine $\underline{\pi} = (\underline{\pi}_0, \underline{\pi}_1, \ldots, \underline{\pi}_b)$. Furthermore, we may restrict the infinitesimal generator matrix to

$$Q = \begin{array}{c|ccccccc|} & 0 & 1 & 2 & \cdots & b-2 & b-1 & b \\ \hline 0 & Q_{0,0} & Q_{0,1} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 1 & Q_{1,0} & Q_{1,1} & Q_{1,2} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 2 & \mathbf{0} & Q_{2,1} & Q_{2,2} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots & \vdots & \vdots \\ b-2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q_{b-2,b-2} & Q_{b-2,b-1} & \mathbf{0} \\ b-1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q_{b-1,b-2} & Q_{b-1,b-1} & Q_{b-1,b} \\ b & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & Q_{b,b-1} & Q_{b,b} \\ \hline \end{array}, \tag{1.36}$$

so Equations (1.32) and (1.32) now become

$$\underline{0} = \underline{\pi}_0 Q_{0,0} + \underline{\pi}_1 Q_{1,0}, \tag{1.37}$$

$$\underline{0} = \underline{\pi}_i Q_{i,i+1} + \underline{\pi}_{i+1} Q_{i+1,i+1} + \underline{\pi}_{i+2} Q_{i+2,i+1}, \; i = 0, 1, \ldots, b-2, \tag{1.38}$$

$$\underline{0} = \underline{\pi}_{b-1} Q_{b-1,b} + \underline{\pi}_b Q_{b,b}. \tag{1.39}$$

From Equations (1.34) and (1.39), or by substituting $R_{b+1} = \mathbf{0}$ into Equation (1.35) for $i = b$, we have

$$R_b = -Q_{b-1,b}(Q_{b,b})^{-1}, \tag{1.40}$$

which we may calculate (as it is only a function of known $Q_{i,j}$ blocks) and use as an initial point to calculate $R_{b-1}, R_{b-2}, \ldots, R_1$ recursively from Equation (1.35).

We can now express all $\underline{\pi}_i$'s in terms of $R_i$ matrices that we can calculate as well as $\underline{\pi}_0$, so if we can solve for the latter, then we have found every steady-state probability of the level-dependent QBD. If we define $R_0 = Q_{0,0} + R_1 Q_{1,0}$, then Equation (1.37) can be rewritten as

$$\underline{\pi}_0 R_0 = \underline{0}. \tag{1.41}$$

Also, the normalization condition becomes

$$1 = \underline{\pi}\,\underline{e}' = \sum_{i=0}^{b} \underline{\pi}_i \underline{e}' = \underline{\pi}_0 \left( I + \sum_{i=1}^{b} \prod_{j=1}^{i} R_j \right) \underline{e}' = \underline{\pi}_0 \underline{u}', \tag{1.42}$$

where we assume each $\underline{e}'$ has an appropriate length to guarantee that the matrix multiplications are well defined. Letting

$$\underline{u}' = \underline{e}' + \sum_{i=1}^{b} \prod_{j=1}^{i} R_j \underline{e}',$$

and combining Equations (1.41) and (1.42), we obtain

$$\underline{\pi}_0 \left[ \begin{array}{cc} R_0 & \underline{u}' \end{array} \right] = \left[ \begin{array}{cc} \underline{0} & 1 \end{array} \right].$$

This linear system has one more equation than the number of elements of $\underline{\pi}_0$, so if we drop a column of $R_0$ (such that the inverse of the remaining columns joined with $\underline{u}'$ exists) and one of the zeroes on the right-hand side, then we may calculate $\underline{\pi}_0$, and therefore each $\underline{\pi}_i$,

$i = 1, 2, \ldots, b$. A note that we may make here is that by placing an upper bound on the state space of the level of the process, if there are finite-many states within each level, then the steady-state distribution will always exist. Even if the drift upwards was higher than the drift downwards (as discussed in the context of level-independent QBDs) for all levels, we will be able to calculate $\underline{\pi}$ for a given truncation level $b$. This is an immediate result of the CTMC describing the queueing model having finite-many states. Of course, in this case, it should be clear that the use of the truncated version of that model would be inappropriate as the results would involve potentially huge negative bias in calculations of key statistics, such as the mean queue length (which may not even be finite), or mean sojourn time for an arbitrary customer (defined as their time spent waiting in the queue, plus their time in service). Discussion concerning the act of truncating infinite buffer queueing systems, as well as a technique to reduce negative bias in expected queue lengths, may be found in Chapters 5 and 6 of this thesis.

We now consider the above results in the context of the $M/PH/1+M$ queue. To obtain the steady-state probabilities for a given truncation level $b$, the above algorithm can be applied with

$$\begin{aligned}
& & Q_{0,0} &= -\lambda, & Q_{0,1} &= \lambda\underline{\alpha}_0^*, \\
Q_{1,0} &= \underline{S}'_0, & Q_{1,1} &= S - \lambda I_k, & Q_{1,2} &= \lambda I_k, \\
Q_{i,i-1} &= \underline{S}'_0\,\underline{\alpha}_0^* + (i-1)\gamma I_k, & Q_{i,i} &= S - (\lambda + (i-1)\gamma)I_k, & Q_{i,i+1} &= \lambda I_k,\ i = 1, 2, \ldots, b-1, \\
Q_{b,b-1} &= \underline{b}'_0\,\underline{\alpha}_0^* + (b-1)\gamma I_k, & Q_{b,b} &= S - (b-1)\gamma I_k,
\end{aligned}$$

and

$$\begin{aligned}
R_b &= -\lambda\left(S - (b-1)\gamma I_k\right)^{-1}, \\
R_i &= -\lambda\left(S - (\lambda + (i-1)\gamma)I_k + R_{i+1}(\underline{S}'_0\,\underline{\alpha}_0^* + i\gamma I_k)\right)^{-1},\ i = 1, 2, \ldots, b-1, \\
R_0 &= -\lambda + R_1\underline{S}'_0.
\end{aligned}$$

Assuming that $\gamma > 0$, it must hold that $\lim_{i\to\infty} R_i = \mathbf{0}$. Therefore, even without the truncation the CTMC will always be stable. Note that by truncating at level $b$, we are in actuality approximating the $M/PH/1+M$ queue by a $M/PH/1/b+M$ queue, where the '$b$' denotes the number of spaces in the system (also referred to as the *buffer* of the queue). If we let $b \to \infty$, then this approximation will converge to the true $M/PH/1+M$ queue.

If we reduce the queueing system to $M/M/1+M$ by letting the phase-type service distribution be Exp$(\mu)$ (i.e., PH$_1(\underline{\alpha}_0^* = 1, S = -\mu)$), the above reduces to

$$\begin{aligned}
& & Q_{0,0} &= -\lambda, & Q_{0,1} &= \lambda, \\
Q_{1,0} &= \mu, & Q_{1,1} &= -(\lambda + \mu), & Q_{1,2} &= \lambda, \\
Q_{i,i-1} &= \mu + (i-1)\gamma, & Q_{i,i} &= -(\lambda + \mu + (i-1)\gamma), & Q_{i,i+1} &= \lambda,\ i = 1, 2, \ldots, b-1, \\
Q_{b,b-1} &= \mu + (b-1)\gamma, & Q_{b,b} &= -(\mu + (b-1)\gamma),
\end{aligned}$$

and

$$\begin{aligned}
R_b &= \lambda(\mu + (b-1)\gamma)^{-1}, \\
R_i &= \lambda(\lambda + \mu + (i-1)\gamma - R_{i+1}(\mu + i\gamma))^{-1},\ i = 1, 2, \ldots, b-1, \\
R_0 &= -(\lambda + R_1\mu).
\end{aligned}$$

Solving for $R_{b-1}$, we obtain

$$\begin{aligned}R_{b-1} &= (\mu + (b-2)\gamma)(\lambda + \mu + (b-2)\gamma - R_b(\mu + (b-1)\gamma))^{-1} \\ &= \lambda(\lambda + \mu + (b-2)\gamma - \lambda(\mu + (b-1)\gamma)^{-1}(\mu + (b-1)\gamma)^{-1} \\ &= \lambda(\mu + (b-2)\gamma)^{-1}.\end{aligned}$$

Continuing inductively, we can show that

$$R_i = \frac{\lambda}{\mu + (i-1)\gamma},\ i = 1, 2, \ldots, b,$$

and this leads to

$$\pi_i = \pi_0 \prod_{j=1}^{i} \frac{\lambda}{(j-1) + \mu} = \pi_0 \frac{\lambda^i}{\prod_{j=1}^{i}(\mu + (j-1)\gamma)}. \tag{1.43}$$

Applying the normalization condition $\underline{\pi}\,\underline{e}' = \sum_{i=0}^{b} \pi_i = 1$, we require

$$1 = \pi_0 + \sum_{i=1}^{b} \pi_0 \frac{\lambda^i}{\prod_{j=1}^{i}((j-1)\gamma + \mu)},$$

so it must hold that

$$\pi_0 = \frac{1}{1 + \sum_{i=1}^{b} \frac{\lambda^i}{\prod_{j=1}^{i}((j-1)\gamma + \mu)}}, \tag{1.44}$$

which we can now use to solve for $\pi_i$, $i = 1, 2, \ldots, b$ using Equation (1.43). If we let $\gamma = \mu$, then the queue length distribution of the $M/M/1/b{+}M$ system is equivalent to that of an $M/M/b/b$ (or $M/M/\infty/b$) queueing system, where every customer who enters the system immediately receives service. Letting $\gamma = \mu$ in Equations (1.43) and (1.44), we immediately find the $\pi_i$'s of the $M/M/b/b$ model to be

$$\pi_i = \frac{\rho^i}{i! \sum_{i=0}^{b} \frac{\rho^i}{i!}},\ i = 0, 1, \ldots, b,$$

where we let $\rho = \lambda/\mu$.

Finally, if we let $b \to \infty$ in the $M/M/b/b$ queue, we can obtain the steady-state probabilities for the $M/M/\infty$ queue. Since

$$\lim_{b\to\infty} \sum_{i=0}^{b} \frac{\rho^i}{i!} = \sum_{i=0}^{\infty} \frac{\rho^i}{i!} = e^{\rho},$$

we have

$$\pi_i = e^{-\rho}\frac{\rho^i}{i!},\ i \in \mathbb{N},$$

indicating that the steady-state distribution for a $M/M/\infty$ queue is a Poisson distribution with parameter $\rho$.

### 1.2.7 Polling Systems

A more general queueing system may involve $N \in \mathbb{Z}^+$ queues (indexed as $Q_i$, $i = 1, 2, \ldots, N$), where each queue may hold potentially differing types (or classes) of customers. If the server(s) of the system are common to each queue, in that they are responsible for serving customers from every queue in the system, then the queueing system is referred to as a *polling system.* Aside from the distributions and behaviours for each class of customers (e.g., interarrival, service, and impatience time distributions), the unique features that characterize a polling system are:

- **Switchover Times:** The duration of time that it takes a server to move from one queue to another.

- **Polling Order:** The rules used by a server to determine the order in which they visit each queue within the system.

- **Service Discipline:** The rules which determine how many customers in a queue receive service by a server during a visit to that queue.

The combination of a polling system's polling order and service discipline is known as its *service policy*, which ultimately decides which customer in the system receives service at a given time.

The switchover times for a polling system may be zero or non-zero in duration. When they are non-zero, they may be deterministic or follow a distribution which can depend on what queue the server is switching *from*, as well as what queue the server is switching *to.* In our later sections, we consider switchover times having continuous phase-type distributions, whose subgenerator matrices depend on the queue that the server is switching to. It is for this reason that we later use the terminology *switch-in time*, to indicate that the queue the server is switching into is of the most importance (while the previous location of the server may influence the initial probability vector of the distribution). Typically, after a server completes a switchover, if they arrive to an empty queue, then they immediately leave and begin the next switchover. However, more general models may require a minimum *threshold* of customers present in a queue to allow the server to switch to it, even if their current queue is empty (e.g., Avrachenkov et al. [8], Avram and Gómez-Corral [9], Perel and Yechiali [74]).

As we must now track $N$ queue lengths ($X_i$, $i = 1, 2, \ldots, N$), our CTMCs will typically have dimensions no less than $N$. Assuming that we can describe a polling system using a QBD (i.e., the number of customers in a queue may not change by more than 1 in a single transition), the infinitesimal generator matrices for these models may be either level-independent or level-dependent, where it is standard to treat the length of one of the queues as the level of the process. Even if we are able to make use of a level-independent QBD, as the analysis in Section 1.2.4 requires us to calculate $(I - R)^{-1}$ for Equation (1.28), it is necessary for the number of states within any level to be finite. This implies the necessity of finite buffers for all queues other than the one which we treat as the level of the process.

In the case of a level-dependent model, every queue will be required to have a finite buffer. This is the main downside of using MAM to approximate infinite buffer polling models, as the higher dimensionality of tracking multiple queue lengths will further increase the cost of large buffer sizes in terms of system resources (e.g., computer memory). However, the need of high buffers is less if, for example, the system has low traffic or if the customers are impatient, as it becomes increasingly unlikely to observe large queue lengths. Even if this is not the case, computer technology is becoming ever more powerful which constantly improves the accuracy of these models over time, while a method such as the Unobserved Waiting Customer approximation covered in Chapters 5 and 6 of this thesis may also be applied to reduce bias inherent from the use of finite buffers. For example, Stern [87] examined the use of a finite buffer approximation within a $M/M/1$ model's transient analysis, and found it accurate at their chosen buffer for moderate $\rho$, but less so for high $\rho$. Of course, this could be remedied by them somewhat by simply choosing a higher buffer. If a queueing model naturally has finite buffers, or considers a closed queueing system with a finite population of customers who alternate between requiring and not requiring service (e.g., a closed queueing system describing the maintenance of an inventory of machines), then these methods can accurately describe the entire system without losing any states to truncation.

In the review paper of Vishnevskii and Semenova [93], several classic examples of polling orders and service disciplines are listed. We shall briefly touch on them, providing examples on how some of their structures may be implemented within a generator to be used within a matrix analytic framework. They note that a polling order may be static or dynamic in nature, depending on if the rules governing the server's movements between queues are consistent at all times, or if the decision on which queue to visit is made at particular instants of time based on full or partial information about the polling system, respectively. Some examples of static polling orders are:

- **Cyclic Order:** Within a *cycle*, the server visits each queue in order, exactly once. Specifically, the order of visitations is $Q_1, Q_2, \ldots, Q_{N-1}, Q_N, Q_1, Q_2, \ldots$, with a new cycle beginning at the start of each visit to $Q_1$.

- **Periodic Order:** Within a cycle, the server visits queues in a repeating sequence which may involve multiple visits to one or more queues. Specifically, the order of visitations is $Q_{T(1)}, Q_{T(2)}, \ldots, Q_{T(M-1)}, Q_{T(M)}, Q_{T(1)}, Q_{T(2)}, \ldots$, where indexes $\{T(1), T(2), \ldots, T(M)\}$, $M \geq N$, $T(i) \in \{1, 2, \ldots, N\}$, form a *polling table.* Under a periodic order, it is possible to allow multiple visits to one or more queues within a cycle, which repeats after every $M^{\text{th}}$ queue visit.

- **Random Order:** After serving $Q_i$, the server next visits $Q_j$ with probability $p_{i,j}$, $i, j = 1, 2, \ldots, N$, where $\sum_{j=1}^{N} p_{i,j} = 1$, $\forall\ i = 1, 2, \ldots, N$. A simplified version of this polling order may allow for the selection probabilities to be independent of $i$.

- **Priority Order:** Each queue within the system is assigned a priority, and a given queue may only receive service if all higher priority queues are empty of customers. This may take the form of a *preemptive priority* order, where the arrival of a higher priority customer causes an immediate switch by the server, interrupting their service, or a *non-preemptive priority* order, where the server checks for higher priority customers only at service completion time instants, and a customer's service will never be interrupted.

A service discipline may indicate that the number of customers that will receive service during a server's visit to their queue is deterministic, following a set rule, or random, where the number of customers served is a discrete random variable. Some examples of deterministic service disciplines are:

- **Exhaustive Service:** The server continues to serve a queue until it completely empties.

- **Gated Service:** The server only serves the customers who were already in the queue at the beginning of the server's visit. If instead the server treats only customers who were in the queue at the beginning of a cycle, then it is referred to as a *globally-gated* service discipline.

- **$k_i$-Limited Service:** The server will serve up to a maximum of $k_i \in \mathbb{Z}^+$ customers, or until the queue empties.

- **$k_i$-Decrementing Service:** The server will serve customers until the queue length decreases by $k_i \in \mathbb{Z}^+$ relative to its length at the start of the visit, or until the queue empties.

- **Time Limited Service:** The duration of the server's visit to a queue is limited not by a number of customers they may serve, but by a maximum amount of time that they may stay at that queue.

Next, some examples of random service disciplines include:

- **Binomial Service:** (or *Binomial-gated* service) The number of customers that will be served is randomly determined upon the beginning of the server's visit according to a binomial distribution. If the number of customers present at the polling instant at $Q_i$ is $x_i$, and $0 < p_i \leq 1$ is some probability, then the server will serve $n$ class-$i$ customers before leaving with probability

  $$P(\text{Serve } n \text{ class-}i \text{ customers}) = \binom{x_i}{n} p_i^{x_i}(1-p_i)^{x_i - n}, \; n = 0, 1, \ldots, x_i.$$

  If $p_i = 1$, then the binomial discipline reduces to the gated discipline.

- **Bernoulli Service:** After each service completion at $Q_i$, if the queue has not been emptied, the server will serve another customer from $Q_i$ with probability $p_i$. The server will always serve at least one customer (if available), and the probability of serving up to a maximum of $n$ class-$i$ customers during a visit is

  $$P(\text{Serve up to } n \text{ class-}i \text{ customers}) = p_i^{n-1}(1-p_i), \; n \in \mathbb{Z}^+,$$

  which we recognize as a geometric distribution. Such a service policy is convenient as it is memoryless, and so we are not required to track how many customers have received service to maintain the Markov property in a CTMC. If $p_i = 1$, then the Bernoulli discipline reduces to the exhaustive discipline. If $p_i = 0$, then the Bernoulli discipline reduces to the 1-limited discipline.

  **Remark 1.4.** A strategic choice of $p_i$ can allow for the Bernoulli discipline to be comparable to the $k_i$-limited discipline, in that its mean maximum number of services can be set to equal $k_i$. As the mean of a geometric random variable with success parameter $1-p$ is $1/(1-p)$, we can let $p_i = 1 - 1/k_i$ to equate the expected maximum number of services.

We close this subsection by considering some examples of queueing systems involving some of the above policies which will appear later in this thesis within the context of more complicated models. In all examples, we allow for a lone server, $N = 2$ queues, and we assume that customers within the same queue are served in order according to a FCFS discipline. Switchover times are assumed to be independent $\text{Exp}(\tau_i)$ random variables, where $i = 1, 2$ is the index of the queue that the server is switching to. Additionally, we will let class-$i$ interarrival and service times follow independent $\text{Exp}(\lambda_i)$ and $\text{Exp}(\mu_i)$ distributions, respectively, so that

$$\lambda_{x_1,x_2} = \sum_{i=1}^{2}(1 - \delta_{x_i,b_i})\lambda_i$$

is the total effective arrival rate when there are $x_i$ class-$i$ customers, $i = 1, 2$, where $\delta_{i,j}$ is the standard *Kronecker delta function*, defined as

$$\delta_{i,j} = \begin{cases} 1 & \text{, if } i = j, \\ 0 & \text{, if } i \neq j. \end{cases}$$

We denote the buffer of $Q_i$ by $b_i$, and assume that $b_1 = \infty$ and $b_2 < \infty$. Letting the length of $Q_1$ represent the level of the process, we will construct the blocks of Equation (1.19) using $Q_{i,j}$ notation. Specifically, for $m \in \mathbb{N}$, we assume the general structures

$$Q_{m,m} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ b_2-1 \\ b_2 \end{array} \begin{array}{c} \begin{array}{cccccc} 0 & 1 & 2 & \cdots & b_2-1 & b_2 \end{array} \\ \left[\begin{array}{cccccc} Q_{m,m,0} & (UD)_{m,0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ (LD)_{m,1} & Q_{m,m,1} & (UD)_{m,1} & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (LD)_{m,2} & Q_{m,m,2} & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q_{m,m,b_2-1} & (UD)_{m,b_2-1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & (LD)_{m,b_2} & Q_{m,m,b_2} \end{array}\right] \end{array}, \tag{1.45}$$

and for $n = m-1, m+1$, $n \geq 0$,

$$Q_{m,n} = \begin{array}{c} \\ 0 \\ 1 \\ \vdots \\ b_2 \end{array} \begin{array}{c} \begin{array}{cccc} 0 & 1 & \cdots & b_2 \end{array} \\ \left[\begin{array}{cccc} Q_{m,n,0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & Q_{m,n,1} & \ddots & \mathbf{0} \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & Q_{m,n,b_2} \end{array}\right] \end{array}. \tag{1.46}$$

For use in the forthcoming Example 7, let $\underline{e}_i$ ($\underline{0}_i$) denote a row vector of ones (zeroes) having length $i$, and let $\underline{e}_{i,j}$ denote a row vector of zeroes having length $i$, with the exception of the $j^{\text{th}}$ element being equal to one.

**Example 4:**

**Service Policy:** Cyclic polling order with exhaustive service.

**CTMC:** $\{(X_1(t), X_2(t), L(t)), t \geq 0\}$, where $X_i(t)$ denotes the number of customers at $Q_i$, $i = 1, 2$, and $L(t)$ is used to track the location of the server at time $t$, such that $L = 2i - 1$ represents switching into class $i$ and $L = 2i$ represents serving class $i$, $i = 1, 2$.

**State Space:**

$$\mathcal{S} = \{(X_1, X_2, L) : X_1 \in \mathbb{N}, X_2 \in \{0, 1, \ldots, b_2\}, L \in \Omega_L(X_1, X_2)\},$$

where

$$\Omega_L(X_1, X_2) = \begin{cases} \{1,3\} & \text{, if } X_1 = 0,\ X_2 = 0, \\ \{1,3,4\} & \text{, if } X_1 = 0,\ X_2 > 0, \\ \{1,2,3\} & \text{, if } X_1 > 0,\ X_2 = 0, \\ \{1,2,3,4\} & \text{, if } X_1 > 0,\ X_2 > 0. \end{cases}$$

**Ordered Steady-state Probability Row Vectors:** $\underline{\pi} = (\underline{\pi}_0, \underline{\pi}_1, \ldots)$, where $\pi_{m,n,l}$ is the steady-state probability that the CTMC is in state $(m, n, l)$, and:

$$\underline{\pi}_m = (\underline{\pi}_{m,0}, \underline{\pi}_{m,1}, \ldots, \underline{\pi}_{m,b_2}),\ m \in \mathbb{N},$$

$$\underline{\pi}_{0,0} = (\pi_{0,0,1}, \pi_{0,0,3}),$$

$$\underline{\pi}_{0,n} = (\pi_{0,n,1}, \pi_{0,n,3}, \pi_{0,n,4}),\ n \in \{1, 2, \ldots, b_2\},$$

$$\underline{\pi}_{m,0} = (\pi_{m,0,1}, \pi_{m,0,2}, \pi_{m,0,3}),\ m \in \mathbb{Z}^+,$$

$$\underline{\pi}_{m,n} = (\pi_{m,n,1}, \pi_{m,n,2}, \pi_{m,n,3}, \pi_{m,n,4}),\ m \in \mathbb{Z}^+,\ n \in \{1, 2, \ldots, b_2\}.$$

**Level 0 Generator Components:**

$$Q_{0,0,0} = \begin{array}{ccc} & 1 & 3 \\ 1 & -(\lambda_{0,0} + \tau_1) & \tau_1 \\ 3 & \tau_2 & -(\lambda_{0,0} + \tau_2) \end{array},$$

$$Q_{0,0,n} = \begin{array}{cccc} & 1 & 3 & 4 \\ 1 & -(\lambda_{0,n} + \tau_1) & \tau_1 & 0 \\ 3 & 0 & -(\lambda_{0,n} + \tau_2) & \tau_2 \\ 4 & 0 & 0 & -(\lambda_{0,n} + \mu_2) \end{array},\ n \in \{1, 2, \ldots, b_2\},$$

$$(UD)_{0,0} = \begin{array}{cccc} & 1 & 3 & 4 \\ 1 & \lambda_2 & 0 & 0 \\ 3 & 0 & \lambda_2 & 0 \end{array},\ (UD)_{0,n} = \lambda_2 I_3,\ n \in \{1, 2, \ldots, b_2 - 1\},$$

$$(LD)_{0,1} = \begin{array}{ccc} & 1 & 3 \\ 1 & 0 & 0 \\ 3 & 0 & 0 \\ 4 & \mu_2 & 0 \end{array},\ (LD)_{0,n} = \begin{array}{cccc} & 1 & 3 & 4 \\ 1 & 0 & 0 & 0 \\ 3 & 0 & 0 & 0 \\ 4 & 0 & 0 & \mu_2 \end{array},\ n \in \{2, 3, \ldots, b_2\},$$

and

$$Q_{0,1,0} = \begin{array}{cccc} & 1 & 2 & 3 \\ 1 & \lambda_1 & 0 & 0 \\ 3 & 0 & 0 & \lambda_1 \end{array},\ Q_{0,1,n} = \begin{array}{ccccc} & 1 & 2 & 3 & 4 \\ 1 & \lambda_1 & 0 & 0 & 0 \\ 3 & 0 & 0 & \lambda_1 & 0 \\ 4 & 0 & 0 & 0 & \lambda_1 \end{array},\ n \in \{1, 2, \ldots, b_2\}.$$

**Level $m$ Generator Components:** For $m \in \mathbb{Z}^+$,

$$Q_{m,m,0} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \end{array} \!\!\begin{array}{c} \begin{array}{ccc} \quad 1 \quad & \quad 2 \quad & \quad 3 \quad \end{array} \\ \left[\begin{array}{ccc} -(\lambda_{m,0}+\tau_1) & \tau_1 & 0 \\ 0 & -(\lambda_{m,0}+\mu_1) & 0 \\ \tau_2 & 0 & -(\lambda_{m,0}+\tau_2) \end{array}\right] \end{array},$$

$$Q_{m,m,n} = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & -(\lambda_{m,n}+\tau_1) & \tau_1 & 0 & 0 \\ 2 & 0 & -(\lambda_{m,n}+\mu_1) & 0 & 0 \\ 3 & 0 & 0 & -(\lambda_{m,n}+\tau_2) & \tau_2 \\ 4 & 0 & 0 & 0 & -(\lambda_{m,n}+\mu_2) \end{array}, \ n \in \{1,2,\ldots,b_2\},$$

$$(UD)_{m,0} = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & \lambda_2 & 0 & 0 & 0 \\ 2 & 0 & \lambda_2 & 0 & 0 \\ 3 & 0 & 0 & \lambda_2 & 0 \end{array}, \ (UD)_{m,n} = \lambda_2 I_4, \ n \in \{1,2,\ldots,b_2-1\},$$

$$(LD)_{m,1} = \begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 \\ 3 & 0 & 0 & 0 \\ 4 & \mu_2 & 0 & 0 \end{array}, \ (LD)_{m,n} = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 \\ 3 & 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & 0 & \mu_2 \end{array}, \ n \in \{2,3,\ldots,b_2\},$$

$$Q_{m,m+1,0} = \lambda_1 I_3, \ Q_{m,m+1,n} = \lambda_1 I_4, \ n \in \{1,2,\ldots,b_2\},$$

$$Q_{1,0,0} = \begin{array}{c|cc} & 1 & 3 \\ \hline 1 & 0 & 0 \\ 2 & 0 & \mu_1 \\ 3 & 0 & 0 \end{array}, \ Q_{1,0,n} = \begin{array}{c|ccc} & 1 & 3 & 4 \\ \hline 1 & 0 & 0 & 0 \\ 2 & 0 & \mu_1 & 0 \\ 3 & 0 & 0 & 0 \\ 4 & 0 & 0 & 0 \end{array}, \ n \in \{1,2,\ldots,b_2\},$$

and for $m = 2,3,\ldots,$

$$Q_{m,m-1,0} = \begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & 0 & 0 & 0 \\ 2 & 0 & \mu_1 & 0 \\ 3 & 0 & 0 & 0 \end{array}, \ Q_{m,m-1,n} = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 0 & 0 & 0 \\ 2 & 0 & \mu_1 & 0 & 0 \\ 3 & 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & 0 & 0 \end{array}, \ n \in \{1,2,\ldots,b_2\}.$$

**Example 5:**

**Service Policy:** Class-1 preemptive priority, where the server will only serve class-2 customers if $X_1(t) = 0$, and will immediately begin a switchover to $Q_1$ if a class-1 arrival is observed. If the server arrives to a queue and finds it empty, they immediately leave to switch to the other queue.

**CTMC:** $\{(X_1(t), X_2(t), L(t)), t \geq 0\}$, where $X_i(t)$ and $L(t)$ are as defined in Example 4.

**State Space:**

$$\mathcal{S} = \{(X_1, X_2, L) : X_1 \in \mathbb{N}, X_2 \in \{0, 1, \ldots, b_2\}, L \in \Omega_L(X_1, X_2)\},$$

where

$$\Omega_L(X_1, X_2) = \begin{cases} \{1,3\} & , \text{ if } X_1 = 0,\ X_2 = 0, \\ \{1,3,4\} & , \text{ if } X_1 = 0,\ X_2 > 0, \\ \{1,2\} & , \text{ if } X_1 > 0. \end{cases}$$

**Ordered Steady-state Probability Row Vectors:** $\underline{\pi} = (\underline{\pi}_0, \underline{\pi}_1, \ldots)$, where $\pi_{m,n,l}$ is the steady-state probability that the CTMC is in state $(m, n, l)$, and:

$$\underline{\pi}_m = (\underline{\pi}_{m,0}, \underline{\pi}_{m,1}, \ldots, \underline{\pi}_{m,b_2}),\ m \in \mathbb{N},$$

$$\underline{\pi}_{0,0} = (\pi_{0,0,1}, \pi_{0,0,3}),$$

$$\underline{\pi}_{0,n} = (\pi_{0,n,1}, \pi_{0,n,3}, \pi_{0,n,4}),\ n \in \{1, 2, \ldots, b_2\},$$

$$\underline{\pi}_{m,n} = (\pi_{m,n,1}, \pi_{m,n,2}),\ m \in \mathbb{Z}^+,\ n \in \{0, 1, \ldots, b_2\}.$$

**Level 0 Generator Components:** When $X_1(t) = 0$, this queue behaves identically to the cyclic exhaustive queue from Example 4, and these components are the same with the exception of

$$Q_{0,1,0} = \begin{array}{c} \\ 1 \\ 3 \end{array}\!\!\begin{array}{c} \begin{array}{cc} 1 & 2 \end{array} \\ \begin{bmatrix} \lambda_1 & 0 \\ \lambda_1 & 0 \end{bmatrix} \end{array},\ Q_{0,1,n} = \begin{array}{c} \\ 1 \\ 3 \\ 4 \end{array}\!\!\begin{array}{c} \begin{array}{cc} 1 & 2 \end{array} \\ \begin{bmatrix} \lambda_1 & 0 \\ \lambda_1 & 0 \\ \lambda_1 & 0 \end{bmatrix} \end{array},\ n \in \{1, 2, \ldots, b_2\}.$$

**Level $m$ Generator Components:** For $m \in \mathbb{Z}^+$,

$$Q_{m,m,n} = \begin{array}{c|cc} & 1 & 2 \\ \hline 1 & -(\lambda_{m,n} + \tau_1) & \tau_1 \\ 2 & 0 & -(\lambda_{m,n} + \mu_1) \end{array},\ n \in \{0, 1, \ldots, b_2\},$$

$$(UD)_{m,n} = \lambda_2 I_2,\ n \in \{0, 1, \ldots, b_2 - 1\},$$

$$(LD)_{m,n} = \mathbf{0},\ n \in \{1, 2, \ldots, b_2\},$$

$$Q_{m,m+1,n} = \lambda_1 I_2,\ n \in \{0, 1, \ldots, b_2\},$$

$$Q_{1,0,0} = \begin{array}{c|cc} & 1 & 3 \\ \hline 1 & 0 & 0 \\ 2 & 0 & \mu_1 \end{array},\ Q_{1,0,n} = \begin{array}{c|ccc} & 1 & 3 & 4 \\ \hline 1 & 0 & 0 & 0 \\ 2 & 0 & \mu_1 & 0 \end{array},\ n \in \{1, 2, \ldots, b_2\},$$

and for $m = 2, 3, \ldots$,

$$Q_{m,m-1,n} = \begin{array}{c|cc} & 1 & 2 \\ \hline 1 & 0 & 0 \\ 2 & 0 & \mu_1 \end{array},\ n \in \{0, 1, \ldots, b_2\}.$$

**Example 6:**

**Service Policy:** Cyclic polling order with *smart Bernoulli* service, where after each class-$i$ service completion, should the other queue be non-empty, the server begins another service at

$Q_i$ with probability $p_i$, or begins a switchover with probability $1 - p_i$. This specific variation on the normal Bernoulli service discipline prevents the server from leaving customers to go visit a currently empty queue.

**CTMC:** $\{(X_1(t), X_2(t), L(t)), t \geq 0\}$, where $X_i(t)$ and $L(t)$ are as defined in Example 4.

**State Space:**

$$\mathcal{S} = \{(X_1, X_2, L) : X_1 \in \mathbb{N}, X_2 \in \{0, 1, \ldots, b_2\}, L \in \Omega_L(X_1, X_2)\},$$

where $\Omega_L(X_1, X_2)$ is as defined in Example 4.

**Ordered Steady-state Probability Row Vectors:** $\underline{\pi} = (\underline{\pi}_0, \underline{\pi}_1, \ldots)$, where $\pi_{m,n,l}$ is the steady-state probability that the CTMC is in state $(m, n, l)$, and each $\underline{\pi}_m$ is as defined in Example 4.

**Level** 0 **Generator Components:** When either queue is empty, the smart Bernoulli service discipline acts the same as the exhaustive service discipline, so these components are the same as those in Example 4.

**Level** $m$ **Generator Components:** For $m \in \mathbb{Z}^+$, all components are the same as those in Example 4 with the exception of

$$(LD)_{m,n} = \begin{array}{c c} & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ (1-p_2)\mu_2 & 0 & 0 & p_2\mu_2 \end{bmatrix} \end{array}, \; n \in \{2, 3, \ldots, b_2\},$$

and for $m = 2, 3, \ldots$,

$$Q_{m,m-1,0} = \begin{array}{c c} & \begin{array}{ccc} 1 & 2 & 3 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \end{array} & \begin{bmatrix} 0 & 0 & 0 \\ 0 & \mu_1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \end{array}, \; Q_{m,m-1,n} = \begin{array}{c c} & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & p_1\mu_1 & (1-p_1)\mu_1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}, \; n \in \{1, 2, \ldots, b_2\}.$$

If we let $p_i = 1$, $i = 1, 2$, then these components will also become identical to those from Example 4, as the smart Bernoulli service policy becomes the exhaustive service policy.

**Example 7:**

**Service Policy:** Cyclic polling order with $k_i$-limited service.

**CTMC:** $\{(X_1(t), X_2(t), L(t), K(t)), t \geq 0\}$, where $X_i(t)$ and $L(t)$ are as defined in Example 4, and $K(t) = k \in \mathbb{Z}^+$ implies that the server is on their $k^{\text{th}}$ service within a visit to a queue, while we let $K(t) = 0$ when the server is switching between queues.

**State Space:**

$$\mathcal{S} = \{(X_1, X_2, L, K) : X_1 \in \mathbb{N}, X_2 \in \{0, 1, \ldots, b_2\}, L \in \Omega_L(X_1, X_2), K \in \Omega_K(L)\},$$

where $\Omega_L(X_1, X_2)$ is as defined in Example 4, and

$$\Omega_K(L) = \begin{cases} \{0\} & , \text{ if } L = 1, \\ \{1, 2, \ldots, k_1\} & , \text{ if } L = 2, \\ \{0\} & , \text{ if } L = 3, \\ \{1, 2, \ldots, k_2\} & , \text{ if } L = 4. \end{cases}$$

**Ordered Steady-state Probability Row Vectors:** $\underline{\pi} = (\underline{\pi}_0, \underline{\pi}_1, \ldots)$, where $\pi_{m,n,l,k}$ is the steady-state probability that the CTMC is in state $(m, n, l, k)$, and:

$$\underline{\pi}_m = (\underline{\pi}_{m,0}, \underline{\pi}_{m,1}, \ldots, \underline{\pi}_{m,b_2}),\ m \in \mathbb{N},$$

$$\underline{\pi}_{0,0} = (\pi_{0,0,1,0}, \pi_{0,0,3,0}),$$

$$\underline{\pi}_{0,n} = (\pi_{0,n,1,0}, \pi_{0,n,3,0}, \pi_{0,n,4,1}, \pi_{0,n,4,2}, \ldots, \pi_{0,n,4,k_2}),\ n \in \{1, 2, \ldots, b_2\},$$

$$\underline{\pi}_{m,0} = (\pi_{m,0,1,0}, \pi_{m,0,2,1}, \pi_{0,n,2,2}, \ldots, \pi_{0,n,2,k_1} \pi_{m,0,3,0}),\ m \in \mathbb{Z}^+,$$

$$\begin{aligned} \underline{\pi}_{m,n} = (&\pi_{m,n,1,0}, \pi_{m,n,2,1}, \pi_{m,n,2,2}, \ldots, \pi_{m,n,2,k_1}, \\ &\pi_{m,n,3,0}, \pi_{m,n,4,1}, \pi_{m,n,4,2}, \ldots, \pi_{m,n,4,k_2}),\ m \in \mathbb{Z}^+,\ n \in \{1, 2, \ldots, b_2\}. \end{aligned}$$

**Level 0 Generator Components:**

$$Q_{0,0,0} = \begin{array}{c} \\ 1 \\ 3 \end{array} \!\!\begin{array}{c} \begin{array}{cc} \quad 1 \quad & \quad 3 \quad \end{array} \\ \left[\begin{array}{cc} -(\lambda_{0,0} + \tau_1) & \tau_1 \\ \tau_2 & -(\lambda_{0,0} + \tau_2) \end{array}\right] \end{array},$$

$$Q_{0,0,n} = \begin{array}{c|ccc} & 1 & 3 & 4 \\ \hline 1 & -(\lambda_{0,n} + \tau_1) & \tau_1 & \underline{0}_{k_2} \\ 3 & 0 & -(\lambda_{0,n} + \tau_2) & \tau_2 \underline{e}_{k_2,1} \\ 4 & \underline{0}'_{k_2} & \underline{0}'_{k_2} & -(\lambda_{0,n} + \mu_2) I_{k_2} \end{array},\ n \in \{1, 2, \ldots, b_2\},$$

$$(UD)_{0,0} = \begin{array}{c|ccc} & 1 & 3 & 4 \\ \hline 1 & \lambda_2 & 0 & \underline{0}_{k_2} \\ 3 & 0 & \lambda_2 & \underline{0}_{k_2} \end{array},\ (UD)_{0,n} = \lambda_2 I_{2+k_2},\ n \in \{1, 2, \ldots, b_2 - 1\},$$

$$(LD)_{0,1} = \begin{array}{c|cc} & 1 & 3 \\ \hline 1 & 0 & 0 \\ 3 & 0 & 0 \\ 4 & \mu_2 \underline{e}'_{k_2} & \underline{0}'_{k_2} \end{array},\ (LD)_{0,n} = \begin{array}{c|ccc} & 1 & 3 & 4 \\ \hline 1 & 0 & 0 & \underline{0}_{k_2} \\ 3 & 0 & 0 & \underline{0}_{k_2} \\ 4 & \mu_2 \underline{e}'_{k_2,k_2} & \underline{0}'_{k_2} & D_2 \end{array},\ n \in \{2, 3, \ldots, b_2\},$$

where for $i = 1, 2$,

$$D_i = \begin{cases} 0 & , \text{ if } k_i = 1, \\ \begin{bmatrix} \underline{0}'_{k_i-1} & \mu_i I_{k_i-1} \\ 0 & \underline{0}_{k_i-1} \end{bmatrix} & , \text{ if } k_i > 1, \end{cases}$$

and

$$Q_{0,1,0} = \begin{array}{cccc} & 1 & 2 & 3 \\ 1 & \lambda_1 & \underline{0}_{k_1} & 0 \\ 3 & 0 & \underline{0}_{k_1} & \lambda_1 \end{array}, \ Q_{0,1,n} = \begin{array}{ccccc} & 1 & 2 & 3 & 4 \\ 1 & \lambda_1 & \underline{0}_{k_1} & 0 & \underline{0}_{k_2} \\ 3 & 0 & \underline{0}_{k_1} & \lambda_1 & \underline{0}_{k_2} \\ 4 & \underline{0}'_{k_2} & \mathbf{0} & \underline{0}'_{k_2} & \lambda_1 I_{k_2} \end{array}, \ n \in \{1, 2, \ldots, b_2\}.$$

**Level $m$ Generator Components:** For $m \in \mathbb{Z}^+$,

$$Q_{m,m,0} = \begin{array}{cccc} & 1 & 2 & 3 \\ 1 & -(\lambda_{m,0} + \tau_1) & \tau_1 \underline{e}_{k_1,1} & 0 \\ 2 & \underline{0}'_{k_1} & -(\lambda_{m,0} + \mu_1) I_{k_1} & \underline{0}'_{k_1} \\ 3 & \tau_2 & \underline{0}_{k_1} & -(\lambda_{m,0} + \tau_2) \end{array},$$

$$Q_{m,m,n} = \begin{array}{ccccc} & 1 & 2 & 3 & 4 \\ 1 & -(\lambda_{m,n} + \tau_1) & \tau_1 \underline{e}_{k_1,1} & 0 & \underline{0}_{k_2} \\ 2 & \underline{0}'_{k_1} & -(\lambda_{m,n} + \mu_1) I_{k_1} & \underline{0}'_{k_1} & \mathbf{0} \\ 3 & 0 & \underline{0} & -(\lambda_{m,n} + \tau_2) & \tau_2 \underline{e}_{k_2,1} \\ 4 & \underline{0}'_{k_2} & \mathbf{0} & \underline{0}'_{k_2} & -(\lambda_{m,n} + \mu_2) I_{k_2} \end{array}, \ n \in \{1, 2, \ldots, b_2\},$$

$$(UD)_{m,0} = \begin{array}{ccccc} & 1 & 2 & 3 & 4 \\ 1 & \lambda_2 & \underline{0}_{k_1} & 0 & \underline{0}_{k_2} \\ 2 & \underline{0}'_{k_1} & \lambda_2 I_{k_1} & \underline{0}'_{k_1} & \mathbf{0} \\ 3 & 0 & \underline{0}_{k_1} & \lambda_2 & \underline{0}_{k_2} \end{array}, \ (UD)_{m,n} = \lambda_2 I_{2+k_1+k_2}, \ n \in \{1, 2, \ldots, b_2 - 1\},$$

$$(LD)_{m,1} = \begin{array}{cccc} & 1 & 2 & 3 \\ 1 & 0 & \underline{0}_{k_1} & 0 \\ 2 & \underline{0}'_{k_1} & \mathbf{0} & \underline{0}'_{k_1} \\ 3 & 0 & \underline{0}_{k_1} & 0 \\ 4 & \mu_2 \underline{e}'_{k_2} & \mathbf{0} & \underline{0}'_{k_2} \end{array}, \ (LD)_{m,n} = \begin{array}{ccccc} & 1 & 2 & 3 & 4 \\ 1 & 0 & \underline{0}_{k_1} & 0 & \underline{0}_{k_2} \\ 2 & \underline{0}'_{k_1} & \mathbf{0} & \underline{0}'_{k_1} & \mathbf{0} \\ 3 & 0 & \underline{0}_{k_1} & 0 & \underline{0}_{k_2} \\ 4 & \mu_2 \underline{e}'_{k_2,k_2} & \mathbf{0} & \underline{0}'_{k_2} & D_2 \end{array}, \ n \in \{2, 3, \ldots, b_2\},$$

$$Q_{m,m+1,0} = \lambda_1 I_{2+k_1}, \ Q_{m,m+1,n} = \lambda_1 I_{2+k_1+k_2}, \ n \in \{1, 2, \ldots, b_2\},$$

$$Q_{1,0,0} = \begin{array}{ccc} & 1 & 3 \\ 1 & 0 & 0 \\ 2 & \underline{0}'_{k_1} & \mu_1 \underline{e}'_{k_1} \\ 3 & 0 & 0 \end{array}, \ Q_{1,0,n} = \begin{array}{cccc} & 1 & 3 & 4 \\ 1 & 0 & 0 & \underline{0}_{k_2} \\ 2 & \underline{0}'_{k_1} & \mu_1 \underline{e}'_{k_1} & \mathbf{0} \\ 3 & 0 & 0 & \underline{0}_{k_2} \\ 4 & \underline{0}'_{k_2} & \underline{0}'_{k_2} & \mathbf{0} \end{array}, \ n \in \{1, 2, \ldots, b_2\},$$

and for $m = 2, 3, \ldots,$

$$Q_{m,m-1,0} = \begin{array}{cccc} & 1 & 2 & 3 \\ 1 & 0 & \underline{0}_{k_1} & 0 \\ 2 & \underline{0}'_{k_1} & D_1 & \mu_1 \underline{e}'_{k_1,k_1} \\ 3 & 0 & \underline{0}_{k_1} & 0 \end{array}, \ Q_{m,m-1,n} = \begin{array}{ccccc} & 1 & 2 & 3 & 4 \\ 1 & 0 & \underline{0}_{k_1} & 0 & \underline{0}_{k_2} \\ 2 & \underline{0}'_{k_1} & D_1 & \mu_1 \underline{e}'_{k_1,k_1} & \mathbf{0} \\ 3 & 0 & \underline{0}_{k_1} & 0 & \underline{0}_{k_2} \\ 4 & \underline{0}'_{k_2} & \mathbf{0} & \underline{0}'_{k_2} & \mathbf{0} \end{array}, \ n \in \{1, 2, \ldots, b_2\}.$$

### 1.2.8 Discussion of Polling Model Literature

When modelling polling systems, the inherent complexity of tracking arrival and service processes, queue lengths, and the position of server(s) can make exact analysis a tall task. As will be demonstrated within this thesis, MAM may be used to accurately track many variables simultaneously, at the cost of increased dimensionality and computation time. One thing that MAM cannot handle however, is leaving distributions as 'general' (although in practice, the phase-type distributions can be used to approximate a desired non-negative distribution, e.g., Section 3.5.3). When working with generally distributed services, decompositions can sometimes be used to express quantities of interest (e.g., the total work in a system) in terms of easier to analyze models, such as the standard $M/G/1$ queue.

Kleinrock [55] proved for a general class of $M/G/1$ polling systems with zero switchover times, no customer reneging, and no service preemptions (unless the service distributions are exponential, such that they possess the memoryless property), that the total amount of unfinished work (in terms of the amount of time it would take the server to serve all currently queued customers) is independent of the service policy used by the server so long as they cannot interrupt the service of a customer in such a way that progress is lost. This is directly related to the number of customers in each queue, and thus through Little's Law [64], the mean waiting times of these customers. This result paved the way to then equate the weighted sum of mean waiting times to a constant (with respect to the choice of service disciplines at each queue). This result was called the *conservation law*, as it was predicated on the server always working so long as there were one or more customers present in the system to be served (not accumulating work by being idle), or treating the customers in such as way that creates work (in this case, by not interrupting service when it can cause work to be lost), and hence is 'work conserving'. Specifically, this result states that

$$\sum_{n=1}^{N} \rho_i \mathrm{E}[W_i] = \frac{\rho/2}{1-\rho} \sum_{i=1}^{N} \lambda_i \mathrm{E}[Ser_i^2],$$

where we let $Ser_i$, $W_i$, $\lambda_i$, and $\rho_i$ denote class-$i$ service times, waiting times, arrival rates, and workloads, while $\rho = \sum_{i=1}^{N} \rho_i$. This result is important as it shows that any change in service discipline that reduces the waiting time of one class of customer, must come at the cost of increasing the waiting times of one or more other classes.

However, this result does only apply to polling systems which have zero switchover times, the existence of which would necessitate breaking the assumption that the server is always serving a customer when one or more are in the system. In consideration of this, Boxma and Groenendijk [19] developed a unified *pseudo-conservation law* for a subset of service disciplines - namely, exhaustive, gated, 1-limited, and 1-decrementing. They proved that the distribution for the amount of work in a $M/G/1$ cyclic polling system with switchover times can be decomposed into the amount of work in a corresponding $M/G/1$ queue without switchover times at an arbitrary time instant, plus the amount of work in the $M/G/1$ model with switches during a switching period, by applying similar arguments as Fuhrmann and Cooper [34] who themselves decomposed the distribution of the number of customers at customer departure instants in a (non-polling) $M/G/1$ queue with vacations. The work in the $M/G/1$ polling model with switchover times (during switching instants) at an arbitrary time instant during a switch (following service at a particular queue) was broken up into the work that arrived to the most recently visited queue since the server has left, the amount of work in other queues, as well as the amount of work left by the server at that queue. The last of these components depends on the service discipline at a particular queue, and thus must be derived separately for each discipline considered.

As a companion to this work, Boxma and Groenendijk [20] also proved the equivalent result for a discrete-time model with bulk arrivals, which was modified through taking limits as the size of time blocks went to zero and shown to prove the previous continuous-time result, effectively extending it to allow for bulk arrivals (where customer flow is still dictated by a Poisson process, but an 'arrival' can contain one or more customers, for one or more classes). Boxma [18] later more formally considered bulk arrivals in continuous time, while also generalizing from switchover times to more general 'interruptions' in service (under constraints), and providing equations for the mean amount of work left by the server at a queue for more service disciplines, such as reserved gated, Binomial-gated, Binomial-exhaustive, and Bernoulli service.

Boxma [18] also discussed how to estimate the mean waiting times of individual queues when arrivals followed a Poisson process and service was FCFS within each queue, by finding linear relations between these mean waiting times and the residual cycle time for a cycle beginning with a server's visit to that queue. By assuming that the residual cycle times are equal for each queue of origin and substituting these linear equations into the pseudo-conservation law equations, the mean residual cycle time(s) can be found and then used to recover the mean waiting times. Fuhrmann and Wang [35] similarly approximated individual mean waiting times for customers from each queue within a cyclic polling system, after deriving approximate bounds on the pseudo-conservation law for $k_i$-limited service disciplines. Outside of approximating individual mean waiting times, the pseudo-conservation law equations are very useful for testing the accuracy of other approximations, obtaining an understanding on polling model dynamics, and solving for the exact mean waiting times in symmetric models (such that each queue's waiting time distribution is identical).

Of course, conservation and pseudo-conservation laws are far from the only way to analyze a polling model. More recently, Winands et al. [96] considered the methods of *mean value analysis* (MVA) for $M/G/1$ cyclic polling systems with $N$ queues and positive switchover times, using exhaustive and/or gated service disciplines. Rather than deriving the steady-state distribution, their method represents the probability of finding the server at a particular queue or switchover by the ratio of the expected visit or switchover times and the mean cycle duration. Arguments can then be made to relate mean waiting times to functions of these ratios, mean queue lengths (defined for each location of the server, to be solved from a set of $N^2$ linear equations), and mean residual service or switchover times (only requiring knowledge of the first two moments of those generally defined distributions). While this method can solve for the individual mean waiting times for each class of customer given minimal distributional information, it is restricted by limited choices of service discipline, and like the conservation laws, is unable to provide information concerning the density functions (PDF or CDF) of these waiting times.

A common analytical technique found in many papers concerned with $M/G/1$ systems involves *z-transforms* (i.e., probability generating functions, or PGFs) of discrete random variables, such as queue lengths at key time instants, and Laplace-Stieltjes transforms of continuous random variables (e.g., [2, 4, 5, 24, 33, 34, 36, 49, 52, 59, 60, 72, 74, 75, 78, 85, 94, 95, 97, 99, 100]). For a random variable $X$, these are defined respectively as $\mathrm{E}[z^X]$ and $\mathrm{E}[e^{-sX}]$, and are valued for their ease of definition for convolutions of independent random variables, and for the one-to-one uniqueness between a particular distribution and its transform. Unlike MVA or the conservation laws, it is possible to invert the LST of a customer's waiting time or the $z$-transform of a queue length, theoretically or numerically, to recover the steady-state distribution in question. Boon [15] provides an excellent, in-depth look at the application of LST techniques on polling models having service disciplines which satisfy the *branching property*, as examined by Resing [79]:

**Branching Property:** If the server arrives at $Q_i$ to find $k_i$ customers present, then during the course of the server's visit, each of these $k_i$ customers will effectively be replaced in an iid manner by a random population having PGF $h_i(z_1, \ldots, z_N)$, which can be any $N$-dimensional PGF.

In particular, Resing proved a relationship between $M/G/1$ polling models, with either generally distributed switchover times or no switchover times, and multitype branching processes with immigration, in order to find an expression for the joint $z$-transform of all queue lengths at the beginning of a server's visit to a particular queue. Several common service disciplines, such as exhaustive and gated, satisfy this property. However, some disciplines, such as $k_i$-limited and Bernoulli, violate the condition of all customers present at a server's time of visit being replaced (i.e., by those who arrive during the busy period created by their service time in the case of exhaustive, or solely during their service time in the case of gated) and can not be analyzed exactly. Therefore, while the LST methods are very informative for the disciplines they cover, they are not universal.

A niche, but interesting, analytical technique is Blanc's [10] modification of the standard balance equation method as discussed in Section 1.2.2. Based on the *power-series algorithm* (PSA), Blanc showed how one can estimate the steady-state probabilities for queue length vector $\underline{n} = (n_1, \ldots, n_s)$, $p(\rho; \underline{n})$, as power-series expansions as functions of $\rho$ in exponential queueing models that satisfy certain conditions, such as stability, and not allowing all servers to simultaneously idle when there are one or more customers available to be served. That is, within the balance equations, they let

$$p(\rho; \underline{n}) = \rho^{|\underline{n}|} \sum_{k=0}^{\infty} \rho^k u(k; \underline{n}),$$

where $|\underline{n}| = \underline{n}\,\underline{e}'$ is the total number of customers in the system, and then solve for $u(k; \underline{n})$ using a recursive algorithm for as many $k$ as required to meet a desired accuracy. Blanc also discussed a transformation of variable that can be used on $\rho$ when the radius of convergence for the expansions as functions of $\rho$ do not include all $\rho$ such that $|\rho - 0.5| \leq 0.5$, and the calculation of moments was covered. Other than the assumptions required for the power-series expansion to converge, the main limitations of this method are its the dependency on exponential distributions (although it can generalize to handle phase-types, like MAM, by increasing the dimensionality of the model), as well as the reported drastic increase in computation time as the number of queues in a system are increased. However, it is powerful in the sense that specific arrival and departure (e.g., service completions) vectors can be defined for each state to cover unique customer or server behaviours, without worry of needing to find some convenient pattern(s) to help with solving the steady-state probabilities. The method was explicitly applied to cyclic polling models under a Bernoulli service discipline with negligible (i.e., zero) switchover times by Blanc in 1990 [11], and with non-negligible (i.e., positive) switchover times in 1991 [12].

Blanc and van der Mei [14] would go on to apply the PSA method to an optimization problem concerning a multi-queue $M/G/1$ polling model with generally distributed switchover times and Bernoulli service discipline with class-dependent parameters (where the general service and switchover time distributions were approximated by Coxian distributions). Their goal was to minimize a cost function of weighted expected waiting times as a function of the Bernoulli parameters. In order to reduce the dimensionality of the optimization problem, the $c\mu$ rule for priority systems (Meilijson and Yechiali [68]) was used to make an argument for automatically setting Bernoulli parameter $p_i = 1$ for one or more queues that have maximal values of $c_i/(\rho_i/\rho)$, the highest ratio of relative weight (in the cost function, where $\sum_i c_i = 1$) to fraction of total workload. Their logic was that if it is optimal to never serve another queue over the queue in question in a corresponding priority queue with no switchovers, then the server should never want to switch away from serving that queue in this Bernoulli polling model, as long as there are customers present.

The $c\mu$ rule also implies that the server should want to avoid serving the class(es) of customer(s) that minimize this value (when other customers are present). However, when the switchover times are not negligible, the cost of incurring extra idle periods with customers in the system is shown to make this not always optimal, instead desiring positive parameters which are less than 1. We apply Blanc and van der Mei's logic and make similar observations in Section 3.5.3 when optimizing our smart Bernoulli discipline. The $c\mu$ rule was generalized by van Mieghem [91] who investigated a generalized $c\mu$ rule where the cost incurred by a waiting customer can be a non-decreasing convex function of how long long they have been waiting (rather than a flat rate per unit time), and Iravani and Kolfal [47] considered a modified $c\mu$ rule for finite-population queueing systems.

Servi [85] also investigated an optimization problem concerning the choice of Bernoulli parameters in a polling system. They made use of the connection between vacation systems and polling models to apply their previous work (Keilson and Servi [50]) concerning a $GI/G/1$ vacation system where the number of services between vacations was determined by a Bernoulli service discipline, and the polling model with positive switchover times where the server follows a class-dependent Bernoulli discipline at each queue. From the point of view of any one queue, the time spent by the server either switching or serving a different queue can be treated as a vacation, while the visit time to that queue is the busy period between vacation periods. Combining their $GI/G/1$ vacation model work with the $M/G/1$ vacation system decomposition work by Fuhrmann and Cooper [34], as well as their work with Ramaswamy specifically concerning the busy period of an $M/G/1$ vacation model with a Bernoulli service discipline [78], they found the LST of each queue's busy period, and used them to numerically approximate the expected waiting times for each customer class. The choice of Bernoulli parameters allowed this model a unique ability to optimize the overall expected waiting time, by way of selecting parameters to give queues relative priorities. Through a numerical example, this was shown to be very valuable, in that for a two queue system, by giving the class of customer with much longer service times a parameter of less than one (indicating non-exhaustive service), it helped insulate the expected waiting time of the class with shorter service times against an increase in the arrival rate of the class with longer service times.

As seen in Section 1.2.7, there is a connection between Bernoulli and $k_i$-limited service disciplines, in that for either discipline the corresponding parameters can be selected to result in the same expected maximum number of customers served in a single visit. As the choice of $k_i$ is something that can be varied, it is something that can be used for optimization. Borst et al. [17] consider the optimal selection of $k_i$ for $N$ classes of customers in a $M/G/1$ polling system with general switchover times. Specifically, they aimed to minimize the cost function

$\sum_{i=1}^{N} c_i \lambda_i \mathrm{E}[W_i]$ through the selection of vector $\underline{k} = (k_1, \ldots, k_N)$, either unconstrained or subject to $\sum_{i=1}^{N} \gamma_i k_i \leq K$. By letting $\gamma_i = 1$ for $i = 1, \ldots, N$, this sets an upper bound on the total number of services in a cycle, while letting $\gamma_i = \mathrm{E}[Ser_i]$ results in a bound on the total expected service time in a cycle. Due to the complexities of the $k_i$-limited discipline, they apply four approximations for the individual expected waiting times, including, for example, an approximation based on a 1-limited polling table involving $k_i$ separate visitations to queue $i$, an approximation that was the weighted sum of 1-limited and exhaustive approximations, and the approximation derived from a pseudo-conservation law by Fuhrmann and Wang [35]. To test the accuracy of their approximations, Blanc's PSA was used [10, 12, 13]. For the constrained problem, it was always observed that the optimal $k_i$'s satisfied $\underline{k}\,\underline{e}' = K$ (with $\gamma_i = 1$), as the potential benefit of a higher $k_i$ for queue $i$ greatly outweighed the cost to the other $N - 1$ queues. We will make the same observation for our similar model considered in Section 4.5.1, when minimizing the modified cost function

$$\sum_{i=1}^{2} \left\{ c_i \lambda_i \mathrm{E}[W_i^{\#}] + r_i \lambda_i P(\text{Class-}i \text{ customer reneges before reaching service}) \right\},$$

where $W_i^{\#}$ is the time spent waiting in the queue for a class-$i$ customer. For the unconstrained optimization problem, not unlike how the $c\mu$ rule (Meilijson and Yechiali [68]) was applied in the Bernoulli discipline, Borst et al. [17] showed that the queue(s) with the highest value of $c_i/\mathrm{E}[Ser_i]$ should be given the top priority, in the form of $k_i = \infty$ (i.e., they should receive exhaustive service).

Aside from the service discipline, it is also possible to optimize in terms of the polling order within a service policy. In addition to our consideration of the $c\mu$ rule, an example of optimizing a polling order for a network of queues is the work of Browne and Yechiali [24]. They investigated the scheduling of visits to $N$ queues within a cycle, with the goal of minimizing the total duration of a cycle. Both optimization at the start of a cycle, as well as dynamic optimization in the form of selecting the next queue to visit as the server finishes serving the current queue were addressed.

For more information on the study of polling models, the interested reader is recommended to the works of Takagi [89], Levy and Sidi [61], Vishnevskii and Semenova [93], Boon [15] and Boon et al. [16], as well as the numerous references therein.

## 1.3 Main Contributions

Within the remainder of this thesis, we will progress the research of polling models. In particular, we develop several structures, techniques, and approximations that can be used within a MAM framework. In Chapters 2 and 3, we consider the modelling of a finite population maintenance model. Some classic service policies are considered in Chapter 2, and we present a way to model a greatly generalized dynamic server behaviour in Chapter 3 which could be applied to other polling models. Among these policies is $(a, b)$ threshold, which permits a greater flexibility than standard threshold policies, and could be of potential interest to examine in other systems. Within this analysis, in addition to calculating the steady-state distribution, we explore the intricacies of characterizing the sojourn time distribution of a customer in a finite population system. This analysis varies depending on the given service policy, and in Chapter 3 we show how it may be obtained when allowing for the generalized server decision process.

Results concerning the expected number of working machines in this system are derived, including an upper bound that may be reached as the capacity of the system is increased to infinity. It is shown that this limit can be attainable for any service policy when the time to switch between queues is negligible, and we demonstrate how, under the presence of switch-in times, the amount of incurred switches caused by different service policies can impact a system's peak expected performance. This knowledge can be useful for a mechanic responsible for maintaining a large number of machines when determining how to schedule repairs. For example, in a server farm, a 'small' job may involve power cycling a server, while a 'large' job could represent having to replace a piece of hardware. If the time to move from a large job to a small job (and back) is very short, then our observations indicate it would be logical to prioritize completing small jobs and returning them to working order as fast as possible.

In Chapters 4, 5, and 6, we consider polling models whose customer populations are infinite. Due to the limitations of MAM, we are forced to truncate the queue lengths of all but one class when using a level-independent QBD, or all classes when using a level-dependent QBD. MAM enables us to easily incorporate customer impatience in our models, and we demonstrate how customers with reneging rates which depend on their position within their queue may be handled, both in building the infinitesimal generator matrix as well as deriving the distributions of their time spent waiting and the actual waiting times experienced by customers who successfully reach service. Fortunately, the presence of reneging helps keep queue lengths small, however blocking probabilities can still be an issue when modelling a system which in reality has no limits on queue lengths. This issue increases dramatically with the number of queues that we must track at once, which results in very large state spaces. To help illustrate this fact, we develop new recursive structures in Chapters 5 and 6 which enable us to construct an infinitesimal generator matrix for a system with a general number of queues which we may easily adjust. These structures are first considered for a simple exhaustive service policy, but are later generalized to allow for $k_i$-limited or Bernoulli service policies at each queue.

By truncating the state space, any steady-state probability mass corresponding to states above the truncation level will be proportionally redistributed to lower states through normalization. This will have a biasing effect on values such as the expected queue lengths, and result in shorter sojourn times as well. In Chapter 5, we introduce a brand new technique that we call the *Unobserved Waiting Customer* (UWC) approximation. The goal of this approximation is to emulate the presence of customers who may be residing in unobservable positions within their queue (i.e., in positions beyond the truncation level), when the observable portion of said queue is full. If an unobserved waiting customer is present, then when the observed queue length would have decremented due to a customer departure, the unobserved customer would be able to immediately replace them. Through this approximation, we aim to shift excess steady-state probability mass from states below the truncation level, with the goal of bringing their approximate values as close as possible to those of the true infinite buffer model. Logically, this results in excess mass being stored at the truncation level, which is the best case scenario given that we are tracking no higher levels at which we could assign it to.

We derive how to optimally apply this technique to several classic models, including the $M/M/1$, $M/M/1+M$, $M/M/\infty$, and $M/PH/1$ queues. Additionally, two versions of its application to the $M/PH/1+M$ queue are considered and compared, as well as to polling systems containing multiple queues. In Chapter 6, we further generalize one of these versions to allow for our earlier considered queue length dependent reneging rates. The benefit of the UWC approximation is that it can easily be incorporated into any MAM polling model to improve the accuracy of calculated results. For example, while an approximation may be performed to treat a QBD as level-independent beyond some level to allow for infinite buffers on one queue (albeit with a small amount of error experienced due to the approximation that decreases with the level which it is made), this method is in fact limited to one of multiple queues in a polling model. In such a case, UWC may be applied to the other queues within the polling system at no cost in the form of an increased state space.

# Part I : Finite Population Maintenance Models

# Chapter 2

# A 2-Class Maintenance Model with a Finite Population and Competing Exponential Failure Rates

## 2.1 Discussion of Literature

When one considers modelling maintenance systems, polling model service policies may not be the first thing to come to mind. However, in the area of maintenance optimization, deciding what components or systems to repair, and when to repair them, are common queries. In fact, two of the first papers to model systems that we now identify as polling models were regarded as maintenance problems! Mack et al. [66] investigated the efficiency of a closed system of machines, which were serviced by a patrolling repairman who would visit each machine in a cyclic fashion. Mack [65] would go on to revisit and generalize this model, extending the constant repair times to discrete random variables.

When we refer to a model as a maintenance system, it is immediately clear that repairs and/or replacements will be involved. There are, however, very distinct types of models that can claim this label. This depends on what, exactly, is being maintained over time. For example, a model may concern itself with the condition of a central machine, rather than being directly connected to a queueing-related issue. Alfa and Castro [3] derived the steady-state distribution of a discrete-time model of a system consisting of a single machine that was at risk of failing. The machine would have a natural lifespan after which it would automatically fail, or it had the potential to randomly fail after each time increment. A machine could be repaired up to a selected (finite) maximum number of times, after which it would need to be replaced, while it was also possible to suffer a large failure requiring a replacement at an earlier incident. The lifetime of the machine, as well as customer service time distributions, were allowed to depend on the number of times the machine has been repaired since the last replacement. An optimization problem was conducted to select the optimal maximum number of repairs permitted, where the system profited while working, but would incur a loss every time the system was repaired or replaced. This work was similar to that of Neuts et al. [71], who considered a comparable continuous-time model where the failures occurred according to a Poisson process. Pérez-Ocón and Montoro-Cazorla [75] would later expand on the continuous-time model by providing a way to numerically solve for the transition probability function matrices (as functions of time) for each operation and repair state, among other contributions.

When considering maintenance in a queueing system, depending on whether the 'server(s)' or the 'customers' are the ones receiving repairs, the interpretation and analysis of the model will vary greatly. In the former case, the machine(s) we repair may provide a function integral to the service process of customers. When a machine is broken, this imparts costs upon the system in terms of increased customer waiting times as well as increasing the probability of a customer abandoning their queue due to impatience. In the latter case, the server may be a repairman who tends to a closed system of machines or components that 'arrive' to the queueing system by failing, where they will wait to be repaired. This is the type of maintenance system that we analyze in Chapters 2 and 3 of this thesis.

We make use of MAM to analyze our *customer-centric* models. MAM is also a convenient tool for *server-centric* maintenance models. For instance, Yang et al. [99] used matrix analytic methods in their investigation of a queueing system where the server would break down over time according to random shocks modelled by a Poisson process. The magnitude of these shocks were non-negative discrete random variables, which incremented the state of the server by the magnitude of the shock. The server's exponential service rate was inversely related to its state, and at a certain finite state, the server would completely break down and cease to work. Similarly, the exponential rates for the repair times also depended on the state of the server. In the interest of optimization, a lower limit was enforced such that if the state of the server equaled or exceeded this point, repair was started rather than waiting for the server to completely break down. A cost function was defined, attributing holding costs over time to customers waiting in queue, as well as at time instants of repair completion. This model was further generalized by Chakravarthy [27], who introduced a probability of a shock not affecting the server if the server was idle at the time, and replaced the assumption of a Poisson process customer flow by a more flexible Markovian arrival process. Chakravarthy expressed the distribution of several key system characteristics as phase-type distributions, such as the effective service time and repair duration. Further examples of other server-centric maintenance models which do not employ matrix analytic methods include the works of Hsu [45], Perry and Posner [76], and Peschansky and Kovalenko [77].

Clearly, the patrolling repairman models of Mack et al. [66] and Mack [65] are both examples of customer-centric maintenance models. Kim and Koenigsberg [54] considered a system consisting of a server repairing machines on two rotating carousel conveyors. They assumed that the machines had exponentially distributed failure times, while service times as well as the time for an adjacent machine on the same carousel to rotate to the server were constant. This allowed them to apply some of the results from Mack et al. [66]. Both the utilization of the server and the efficiency of the machines were examined. Another example of a customer-centric maintenance model is the work of Righter [80] who investigated a closed queueing system that could function so long as there was at least one working component. Therefore, the system was only down if every single component was either waiting to be repaired or undergoing repair by the single server. Clearly, it was optimal to only have a single component working (and hence, at risk of failing) at a time, but the optimal order (to minimize system downtime) in which they are turned on, and the order in which they are to be repaired (should more than one be down at a time), was investigated.

Finally, we cite the closed queueing model of Gross et al. [42], who considered a closed system of $M + y$ machines, up to $M$ of which could be turned on and working at any time having competing exponential failure times, which may result in either a minor or major repair being required. Every failed machine would either be routed to the minor or major repair node, and those that receive minor repair may still be routed through the major repair node prior to being returned to operation. Each repair node was permitted to have multiple servers in parallel, and the optimal selection of $y$ as well as the number of servers at each queue was investigated. Every distribution was assumed to be exponential, so that the analysis of their system was in the style of Gordon and Newell [38] and Buzen [26] for closed queueing networks with exponential servers. The reason that we single out this paper is that the concept of a closed network of machines which suffer either minor or major failures according to competing exponential failure rates is, in a way, analogous to our maintenance models of interest. We divert, however, in that machines suffering minor failures are never routed through the major failures queue before becoming operational again, and we only have a single server who alternates serving between the two queues according to some specified service policy. Moreover, in Chapter 2 we do not assume the existence of a maintenance float of additional machines that can be functional (but not turned on), however we will consider this extension in Chapter 3. Madu [67] also considered a similar model to Gross et al. [42], differing in that only one machine could be turned on at a time, only a single server was at either repair node, and failed machines always had to initially go through the minor repair node prior to possibly being routed to the major repair node. Abboud [1] later developed an efficient iterative method to find the optimal number of servers and machines for the same model as Gross et al. [42]. We close this subsection by remarking that a majority of the work within this chapter may be found in Granville and Drekic [40].

## 2.2 Model Assumptions

We introduce a maintenance system characterized as a polling model with two classes, each of which represents a different type of failure which may require differently distributed service times to repair by a lone mechanic. Let $C$ be the total number of machines in the system, which are all simultaneously subject to exponential failure rates as long as they are working. Define $\alpha_i$, $i = 1, 2$, to be the exponential rate for class-$i$ failures, so that each machine has a total failure rate of $\alpha = \alpha_1 + \alpha_2$. Once a machine has failed (or arrived to class $i$), it waits in the $i^{\text{th}}$ queue to be served on a FCFS basis amongst other machines in that same queue. It is assumed that only one type of failure can happen to a machine at once, and that the times until failure of each of the machines are independent. While not represented as being in a queue directly, we denote working machines as being of class 0. When the system is empty, the server will move to a location separate from either queue to idle. For notational convenience, we denote the event of the server being idle as the server visiting class 0. Figure 2.1 depicts our maintenance model, where solid black circles represent machines, $X_1$ and $X_2$ are the respective lengths of queues 1 and 2, and $L$ represents the location of the server, to be defined in Section 2.3.

The service policy that the mechanic (henceforth referred to as the server) uses to serve customers from either class may be *exhaustive*, where the server stays at one location and serves that class until its queue empties, or *priority-based*, preferring to serve one class (i.e., the high priority class) over the other (i.e., the low priority class). Among the priority policies, both *non-preemptive* and *preemptive resume* are considered. Under non-preemptive priority, the server immediately switches to serve the high priority class if an arrival is observed while the server is idle or conducting a switch-in time, or after a service completion of the low priority class given that there are high priority customers waiting in their queue. Under preemptive resume priority, the server always switches to serve any high priority customers upon their arrival to the system. If the server happened to be serving a low priority customer at the time

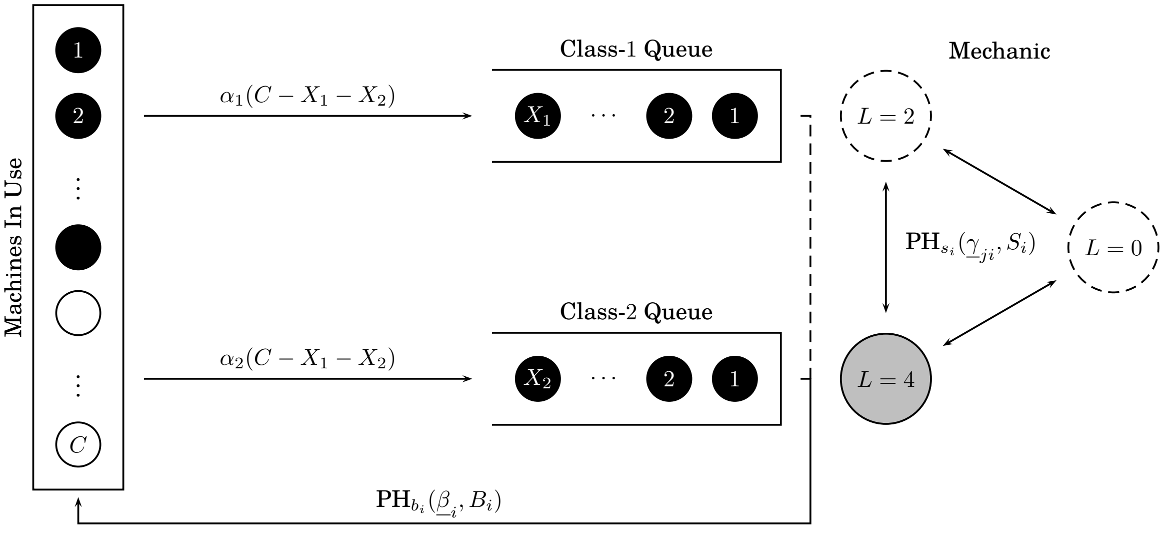
Machines In Use
1
2
⋮
⋮
$C$
$\alpha_1(C - X_1 - X_2)$
$\alpha_2(C - X_1 - X_2)$
Class-1 Queue
$X_1$
⋯
2
1
Class-2 Queue
$X_2$
⋯
2
1
Mechanic
$L = 2$
$L = 0$
$L = 4$
$\mathrm{PH}_{s_i}(\underline{\gamma}_{ji}, S_i)$
$\mathrm{PH}_{b_i}(\underline{\beta}_i, B_i)$

Figure 2.1: Depiction of the maintenance model with the server at queue 2.

of switching, the partially rendered service of the interrupted customer is retained when the server eventually returns after emptying the high priority queue. Let $\mathcal{I}$ denote the type of service policy in place, such that

$$\mathcal{I} = \begin{cases} -2 & \text{, if class 2 has non-preemptive priority over class 1,} \\ -1 & \text{, if class 1 has non-preemptive priority over class 2,} \\ 0 & \text{, if the exhaustive service policy is in place,} \\ 1 & \text{, if class 1 has preemptive resume priority over class 2,} \\ 2 & \text{, if class 2 has preemptive resume priority over class 1.} \end{cases}$$

For $i = 1, 2$, class-$i$ service times are assumed to be non-zero in duration, having a (continuous) phase-type distribution with representation $Ser_i \sim \mathrm{PH}_{b_i}(\underline{\beta}_i, B_i)$. Note that here we are using $B_i$ to denote the rate matrix of a phase-type distribution, not a random variable. Service times are assumed to be independent of each other and of the failure times. Similarly, class-$i$ switch-in times are assumed to have a phase-type distribution with representation $\mathrm{PH}_{s_i}(\underline{\gamma}_{ji}, S_i)$. A class-$i$ switch-in time can be understood as the period of time it takes the server to prepare before beginning work on the class-$i$ queue, after previously attending to something else (e.g., serving customers in the queue of the other class or being idle). We allow the initial probability row vector $\underline{\gamma}_{ji}$ to depend on the class that the server is switching to (i.e., class $i$) and where the server is switching from (i.e., class $j$). We further assume that switch-in times are independent of the service and failure times, as well as the assumption that switching from a switch-in to class $j$ is the same as switching from serving class $j$. For example, if class 1 has higher priority and the server is currently conducting a switch-in to class 2 when a class-1 failure is observed, the initial probability vector $\underline{\gamma}_{21}$ is used for the new switch-in to go serve class 1. In the same way, if the server switches after a class-2 service has completed (as in the case of class 2 emptying, or under non-preemptive priority), or during a class-2 service (as in the case of preemptive resume priority), $\underline{\gamma}_{21}$ is also used. Finally, for the switch-in times, we relax the non-zero duration assumption and let $\gamma_{ji}^{[0]} = 1 - \underline{\gamma}_{ji}\underline{e}'$ be the probability of a switch-in time from class $j$ to class $i$ being zero.

We analyze this model using MAM, representing the system as a level-dependent QBD with the length of the class-1 queue serving as the level of the process. The associated infinitesimal generator is of the form

$$Q^{[C]} = \begin{array}{c|ccccccc|} & 0 & 1 & 2 & \cdots & C-2 & C-1 & C \\ 0 & Q_{0,0}^{[C]} & Q_{0,1}^{[C]} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 1 & Q_{1,0}^{[C]} & Q_{1,1}^{[C]} & Q_{1,2}^{[C]} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 2 & \mathbf{0} & Q_{2,1}^{[C]} & Q_{2,2}^{[C]} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ C-2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q_{C-2,C-2}^{[C]} & Q_{C-2,C-1}^{[C]} & \mathbf{0} \\ C-1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q_{C-1,C-2}^{[C]} & Q_{C-1,C-1}^{[C]} & Q_{C-1,C}^{[C]} \\ C & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & Q_{C,C-1}^{[C]} & Q_{C,C}^{[C]} \end{array}, \tag{2.1}$$

where we recall that $\mathbf{0}$ represents an appropriately dimensioned zero matrix. Note that $Q^{[C]}$ is block-structured, and built in an analogous way to Equation (1.36), in such a way that the submatrices (or blocks) $Q_{i,j}^{[C]}$ contain all transitions where the level changes from $i$ to $j$. The particular forms of these blocks will be specified over the next two subsections for each of the aforementioned service policies. In addition, the superscript $[C]$ of $Q^{[C]}$ (as well as its associated blocks) corresponds to the number of machines in the system that is being modelled, and this choice of notation will be helpful in the upcoming sojourn time analysis. The steady-state distribution for this model may be found using the procedure for level-dependent QBDs as discussed in Section 1.2.6.

## 2.3 Exhaustive and Non-preemptive Priority Service Models

In this subsection, we focus solely on exhaustive and non-preemptive priority service policies (i.e., $\mathcal{I} \in \{-2, -1, 0\}$). As such, we need not consider server movements that interrupt the service of a customer from either class. We may model the system by the CTMC

$$\{(X_1(t), X_2(t), L(t), Y(t)), t \geq 0\},$$

where $X_i(t)$ is the length of the class-$i$ queue, $i = 1, 2$, $L(t) \in \{0, 1, 2, 3, 4, 5\}$ indicates the position of the server (0: server is idle; 1: switch-in to class 1; 2: serving class 1; 3: switch-in to class 2; 4: serving class 2; 5: switch-in to class 0), and $Y(t)$ denotes the phase of the service or switch-in time which has possible values depending on $L(t)$ in the following way:

$$Y(t) \in \Omega_Y(L(t)) = \begin{cases} \{0\} & \text{, if } L(t) = 0, \\ \{1, 2, \ldots, s_1\} & \text{, if } L(t) = 1, \\ \{1, 2, \ldots, b_1\} & \text{, if } L(t) = 2, \\ \{1, 2, \ldots, s_2\} & \text{, if } L(t) = 3, \\ \{1, 2, \ldots, b_2\} & \text{, if } L(t) = 4, \\ \{1, 2, \ldots, s_0\} & \text{, if } L(t) = 5. \end{cases}$$

Let $\pi_{m,n,l,y}$ be the steady-state probability of observing the CTMC in state $(m, n, l, y)$, where $0 \leq m \leq C$, $0 \leq n \leq C - m$, and $l$ and $y$ take values from the respective supports of $L(t)$ and $Y(t)$, above. The ordered steady-state probability row vector for level 0 is

$$\underline{\pi}_0 = (\pi_{0,0,0,0}, \pi_{0,0,5,1}, \ldots, \pi_{0,0,5,s_0}, \underline{\pi}_{0,1}, \ldots, \underline{\pi}_{0,C}),$$

where

$$\underline{\pi}_{0,n} = (\pi_{0,n,3,1}, \ldots, \pi_{0,n,3,s_2}, \pi_{0,n,4,1}, \ldots, \pi_{0,n,4,b_2})$$

is a row vector of length $s_2 + b_2$ for $n = 1, 2, \ldots, C$. For non-zero levels, the $m^{\text{th}}$ steady-state probability row vector is given by

$$\underline{\pi}_m = (\underline{\pi}_{m,0}, \underline{\pi}_{m,1}, \ldots, \underline{\pi}_{m,C-m}), \; m = 1, 2, \ldots, C,$$

where

$$\underline{\pi}_{m,0} = (\pi_{m,0,1,1}, \ldots, \pi_{m,0,1,s_1}, \pi_{m,0,2,1}, \ldots, \pi_{m,0,2,b_1}),$$

and for $n = 1, 2, \ldots, C - m$,

$$\underline{\pi}_{m,n} = (\pi_{m,n,1,1}, \ldots, \pi_{m,n,1,s_1}, \pi_{m,n,2,1}, \ldots, \pi_{m,n,2,b_1}, \pi_{m,n,3,1}, \ldots, \pi_{m,n,3,s_2}, \pi_{m,n,4,1}, \ldots, \pi_{m,n,4,b_2}),$$

which are row vectors of length $s_1 + b_1$ and $s_1 + b_1 + s_2 + b_2$, respectively. Clearly, level 0 has $1 + s_0 + C(s_2 + b_2)$ states, whereas level $m \geq 1$ has $s_1 + b_1 + (C - m)(s_1 + b_1 + s_2 + b_2)$ states.

In order to determine $\underline{\pi}$ using the QBD procedure described in Section 1.2.6, we need only specify the blocks of $Q^{[C]}$ defined in Equation (2.1). In what follows, recall that $\delta_{i,j}$ is the standard Kronecker delta function which equals 1 if $i = j$ and 0 if $i \neq j$, and that $I_i$ is an identity matrix of dimension $i \times i$. Furthermore, let $\underline{B}'_{0,i} = -B_i \underline{e}'$ and $\underline{S}'_{0,i} = -S_i \underline{e}'$ be the absorption rate column vectors corresponding to phase-type representations $\text{PH}_{b_i}(\underline{\beta}_i, B_i)$ and $\text{PH}_{s_i}(\underline{\gamma}_{ji}, S_i)$, respectively. The diagonal blocks of $Q^{[C]}$ can be expressed as

$$Q^{[C]}_{0,0} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C-1 \\ C \end{array} \begin{array}{c} \begin{array}{cccccc} 0 & 1 & 2 & \cdots & C-1 & C \end{array} \\ \left[ \begin{array}{cccccc} \Delta_0^{[C]} & C\alpha_2 \underline{e}' \left( \begin{array}{cc} \underline{\gamma}_{02} & \gamma_{02}^{[0]} \underline{\beta}_2 \end{array} \right) & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \left[ \begin{array}{cc} \underline{0}' & \mathbf{0} \\ \gamma_{20}^{[0]} \underline{B}'_{0,2} & \underline{B}'_{0,2} \underline{\gamma}_{20} \end{array} \right] & \Delta_1^{[C]} & (C-1)\alpha_2 I_{s_2+b_2} & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \Gamma & \Delta_2^{[C]} & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \Delta_{C-1}^{[C]} & \alpha_2 I_{s_2+b_2} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \Gamma & \Delta_C^{[C]} \end{array} \right] \end{array},$$

where

$$\Delta_n^{[C]} = \begin{cases} -C\alpha I_{1+s_0} + \left[ \begin{array}{cc} 0 & \underline{0} \\ \underline{S}'_{0,0} & S_0 \end{array} \right] & \text{, if } n = 0, \\ -(C-n)\alpha I_{s_2+b_2} + \left[ \begin{array}{cc} S_2 & \underline{S}'_{0,2} \underline{\beta}_2 \\ \mathbf{0} & B_2 \end{array} \right] & \text{, if } n = 1, 2, \ldots, C, \end{cases}$$

and

$$\Gamma = \left[ \begin{array}{cc} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \underline{B}'_{0,2} \underline{\beta}_2 \end{array} \right],$$

while for $m = 1, 2, \ldots, C$,

$$Q^{[C]}_{m,m} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C-m-1 \\ C-m \end{array} \begin{array}{c} \begin{array}{cccccc} 0 & 1 & 2 & \cdots & C-m-1 & C-m \end{array} \\ \left[ \begin{array}{cccccc} Q^{[C]}_{m,m,0} & (UD)^{[C]}_{m,0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ (LD)^{[C]}_{m,1} & Q^{[C]}_{m,m,1} & (UD)^{[C]}_{m,1} & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (LD)^{[C]}_{m,2} & Q^{[C]}_{m,m,2} & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q^{[C]}_{m,m,C-m-1} & (UD)^{[C]}_{m,C-m-1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & (LD)^{[C]}_{m,C-m} & Q^{[C]}_{m,m,C-m} \end{array} \right] \end{array},$$

where

$$Q^{[C]}_{m,m,n} = \begin{cases} -(C-m)\alpha I_{s_1+b_1} + \begin{bmatrix} S_1 & \underline{S}'_{0,1}\underline{\beta}_1 \\ \mathbf{0} & B_1 \end{bmatrix} & , \text{ if } n = 0, \\ \\ -(C-m-n)\alpha I_{s_1+b_1+s_2+b_2} + \begin{bmatrix} S_1 & \underline{S}'_{0,1}\underline{\beta}_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & B_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & S_2 & \underline{S}'_{0,2}\underline{\beta}_2 \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & B_2 \end{bmatrix} & , \text{ if } n = 1, 2, \ldots, C-m, \end{cases}$$

$$(UD)^{[C]}_{m,n} = \begin{cases} (C-m)\alpha_2 \begin{bmatrix} (1-\delta_{\mathcal{I},-2})I_{s_1} & \mathbf{0} & \delta_{\mathcal{I},-2}\underline{e}'\underline{\gamma}_{12} & \delta_{\mathcal{I},-2}\gamma^{[0]}_{12}\underline{e}'\underline{\beta}_2 \\ \mathbf{0} & I_{b_1} & \mathbf{0} & \mathbf{0} \end{bmatrix} & , \text{ if } n = 0, \\ \\ (C-m-n)\alpha_2 I_{s_1+b_1+s_2+b_2} & , \text{ if } n = 1, 2, \ldots, C-m-1, \end{cases}$$

and

$$(LD)^{[C]}_{m,n} = \begin{cases} \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \underline{B}'_{0,2}\underline{\gamma}_{21} & \gamma^{[0]}_{21}\underline{B}'_{0,2}\underline{\beta}_1 \end{bmatrix} & , \text{ if } n = 1, \\ \\ \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \delta_{\mathcal{I},-1}\underline{B}'_{0,2}\underline{\gamma}_{21} & \delta_{\mathcal{I},-1}\gamma^{[0]}_{21}\underline{B}'_{0,2}\underline{\beta}_1 & \mathbf{0} & (1-\delta_{\mathcal{I},-1})\underline{B}'_{0,2}\underline{\beta}_2 \end{bmatrix} & , \text{ if } n = 2, 3, \ldots, C-m. \end{cases}$$

With regard to the off-diagonal blocks of $Q^{[C]}$, we first have

$$Q^{[C]}_{m,m+1} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C-m-1 \\ C-m \end{array} \begin{array}{c} \begin{array}{ccccc} 0 & 1 & 2 & \cdots & C-m-1 \end{array} \\ \left[ \begin{array}{ccccc} (C-m)\alpha_1 I_{s_1+b_1} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & Q^{[C]}_{m,m+1,1} & \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & Q^{[C]}_{m,m+1,2} & \ddots & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q^{[C]}_{m,m+1,C-m-1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \end{array} \right] \end{array}$$

for $m = 1, 2, \ldots, C-1$, where

$$Q^{[C]}_{m,m+1,n} = (C-m-n)\alpha_1 I_{s_1+b_1+s_2+b_2},\ n = 1, 2, \ldots, C-m-1.$$

Moreover,

$$Q^{[C]}_{0,1} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C-1 \\ C \end{array} \!\!\! \begin{array}{c} \begin{array}{ccccc} 0 & 1 & 2 & \cdots & C-1 \end{array} \\ \left[ \begin{array}{ccccc} C\alpha_1 \underline{e}' \left( \begin{array}{cc} \underline{\gamma}_{01} & \gamma^{[0]}_{01} \underline{\beta}_1 \end{array} \right) & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & Q^{[C]}_{0,1,1} & \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & Q^{[C]}_{0,1,2} & \ddots & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q^{[C]}_{0,1,C-1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \end{array} \right] \end{array},$$

where

$$Q^{[C]}_{0,1,n} = (C-n)\alpha_1 \left[ \begin{array}{cccc} \delta_{\mathcal{I},-1} \underline{e}' \underline{\gamma}_{21} & \delta_{\mathcal{I},-1} \gamma^{[0]}_{21} \underline{e}' \underline{\beta}_1 & (1-\delta_{\mathcal{I},-1}) I_{s_2} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & I_{b_2} \end{array} \right],\ n = 1, 2, \ldots, C-1,$$

and

$$Q^{[C]}_{1,0} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C-2 \\ C-1 \end{array} \!\!\! \begin{array}{c} \begin{array}{ccccccc} 0 & 1 & 2 & \cdots & C-2 & C-1 & C \end{array} \\ \left[ \begin{array}{ccccccc} \left[ \begin{array}{cc} \underline{0}' & \mathbf{0} \\ \gamma^{[0]}_{10} \underline{B}'_{0,1} & \underline{B}'_{0,1} \underline{\gamma}_{10} \end{array} \right] & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & Q^{\star}_{1,0} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & Q^{\star}_{1,0} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q^{\star}_{1,0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & Q^{\star}_{1,0} & \mathbf{0} \end{array} \right] \end{array},$$

where

$$Q^{\star}_{1,0} = \left[ \begin{array}{cc} \mathbf{0} & \mathbf{0} \\ \underline{B}'_{0,1} \underline{\gamma}_{12} & \gamma^{[0]}_{12} \underline{B}'_{0,1} \underline{\beta}_2 \\ \mathbf{0} & \mathbf{0} \end{array} \right].$$

Finally, for $m = 2, 3, \ldots, C$, the remaining blocks of $Q^{[C]}$ are of the form

$$Q^{[C]}_{m,m-1} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C-m-1 \\ C-m \end{array} \!\!\! \begin{array}{c} \begin{array}{ccccccc} 0 & 1 & 2 & \cdots & C-m-1 & C-m & C-m+1 \end{array} \\ \left[ \begin{array}{ccccccc} Q^{[C]}_{m,m-1,0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & Q^{[C]}_{m,m-1,1} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & Q^{[C]}_{m,m-1,2} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q^{[C]}_{m,m-1,C-m-1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & Q^{[C]}_{m,m-1,C-m} & \mathbf{0} \end{array} \right] \end{array},$$

where

$$Q^{[C]}_{m,m-1,n} = \begin{cases} \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \underline{B}'_{0,1}\underline{\beta}_1 \end{bmatrix} & \text{, if } n = 0, \\ \\ \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (1-\delta_{\mathcal{I},-2})\underline{B}'_{0,1}\underline{\beta}_1 & \delta_{\mathcal{I},-2}\underline{B}'_{0,1}\underline{\gamma}_{12} & \delta_{\mathcal{I},-2}\gamma^{[0]}_{12}\underline{B}'_{0,1}\underline{\beta}_2 \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} & \text{, if } n = 1, 2, \ldots, C-m. \end{cases}$$

Next, we turn our attention to deriving the class-1 sojourn time distribution of a broken machine, representing the time between when a machine suffers a class-1 failure and when it is up and working again. To do so, we require the steady-state distribution of the system immediately prior to a class-1 failure. Letting $C_{1,h}$ denote the event of observing a single class-1 failure within the next $h$ time units and $S_{m,n,l,y}$ denote the event that $(X_1(t), X_2(t), L(t), Y(t)) = (m, n, l, y)$ at steady state (such that $P(S_{m,n,l,y}) = \pi_{m,n,l,y}$), it follows that (e.g., Lakatos et al. [58], Chapter 9)

$$\begin{aligned} q_{m,n,l,y} &= P\left((X_1(t), X_2(t), L(t), Y(t)) = (m, n, l, y) \text{ immediately prior to a class-1 failure}\right) \\ &= \lim_{h \to 0} P(S_{m,n,l,y} | C_{1,h}) \\ &= \lim_{h \to 0} \frac{P(C_{1,h}|S_{m,n,l,y})P(S_{m,n,l,y})}{\sum_{x_1}\sum_{x_2}\sum_{w}\sum_{z} P(C_{1,h}|S_{x_1,x_2,w,z})P(S_{x_1,x_2,w,z})} \\ &= \lim_{h \to 0} \frac{(\alpha_1(C-m-n)h + o(h))\pi_{m,n,l,y}}{\sum_{x_1}\sum_{x_2}\sum_{w}\sum_{z}(\alpha_1(C-x_1-x_2)h + o(h))\pi_{x_1,x_2,w,z}} \\ &= \lim_{h \to 0} \frac{\alpha_1(C-m-n)\pi_{m,n,l,y} + o(h)/h}{\sum_{x_1}\sum_{x_2}\sum_{w}\sum_{z} \alpha_1(C-x_1-x_2)\pi_{x_1,x_2,w,z} + o(h)/h} \\ &= \frac{(C-m-n)\pi_{m,n,l,y}}{\sum_{x_1}\sum_{x_2}\sum_{w}\sum_{z}(C-x_1-x_2)\pi_{x_1,x_2,w,z}} \, . \end{aligned} \tag{2.2}$$

Hence, it follows that these probabilities are simply the normalized steady-state class-1 failure rates. Note that the right-hand side of Equation (2.2) equals zero for all $l$ and $y$ when $m+n = C$, as this corresponds to states where every machine has already suffered a failure (and so there are no working machines available to fail).

We must also consider the impact that the arrival may have on the server, should the arrival be to an empty class-1 queue. This distinction is important, since as we can see by contrasting the blocks $Q^{[C]}_{0,1}$ and $Q^{[C]}_{i,i+1}$, only an arrival to an empty queue may trigger the server to move (causing a change in $L(t)$), as any additional arrivals to a non-empty queue simply increments $X_1(t)$ by 1. For either possible service policy, if $L(t) \in \{0, 5\}$ (i.e., the server is idle or switching into the idle state), then the server will immediately begin a switch-in to serve the class-1 arrival. Let

$$q_{0,0,\bullet,\bullet} = q_{0,0,0,0} + \sum_{i=1}^{s_0} q_{0,0,5,i}$$

be the probability of the system being in any of these states immediately prior to the class-1 arrival. Furthermore, if $L(t) = 3$ (i.e., the server is conducting a class-2 switch-in), then the server will similarly initiate a switch to serve the class-1 arrival only when $\mathcal{I} = -1$. As such, let

$$q_{0,+,3,\bullet} = \delta_{\mathcal{I},-1} \sum_{n=1}^{C-1} \sum_{y=1}^{s_2} q_{0,n,3,y}$$

represent the desired probability that $L(t) = 3$ immediately before the class-1 arrival.

In order to construct the distribution of the waiting time (to reach service), we consider how long it takes for the queue in front of the target customer to empty, as well as the duration of time (if any) required for the server to switch to the target customer once at the head of their queue. Since we are considering an arrival to the system, the state of the process immediately prior to the arrival cannot possibly be one with $X_1(t) + X_2(t) = C$, as there would have had to be at least one machine working to fail. Thus, we construct initial probability vectors in the style of a queue featuring $C - 1$ total machines.

We begin by considering the system with $X_1 = 0$ prior to the arrival. Let

$$\underline{q}_{0,n} = ((1 - \delta_{\mathcal{I},-1})q_{0,n,3,1}, \ldots, (1 - \delta_{\mathcal{I},-1})q_{0,n,3,s_2}, q_{0,n,4,1}, \ldots, q_{0,n,4,b_2}) \tag{2.3}$$

be a row vector of length $s_2 + b_2$ corresponding to the possible states when $X_1(t) = 0$ and $X_2(t) = n$, $0 \leq n \leq C-1$. Since we have extracted the probability $q_{0,+,3,\bullet}$ when $\mathcal{I} = -1$, we must remove the probabilities of starting in the states where $L(t) = 3$ (in the class-1 non-preemptive priority case). When $X_1(t) = X_2(t) = 0$, it follows that the only possible states immediately after the class-1 arrival correspond to a class-1 switch-in, which when finished (if not interrupted by a class-2 arrival, should $\mathcal{I} = -2$), leads to the completion of the waiting time. The initial probabilities for these states are contained in the row vector $q_{0,0,\bullet,\bullet}\underline{\gamma}_{01} + q_{0,+,3,\bullet}\underline{\gamma}_{21}$, which when combined with $\underline{q}_{0,n}$ in Equation (2.3), allow us to construct the full initial probability vector when $X_1(t) = 0$, namely

$$\underline{q}_0 = (q_{0,0,\bullet,\bullet}\underline{\gamma}_{01} + q_{0,+,3,\bullet}\underline{\gamma}_{21}, \underline{q}_{0,1}, \ldots, \underline{q}_{0,C-1}),$$

which has length $s_1 + (C - 1)(s_2 + b_2)$.

When $X_1(t) = m \geq 1$ prior to the arrival, there is no shifting of probability mass required. We can simply construct $\underline{q}_m$ in a way which is analogous to how we originally defined $\underline{\pi}_m$ (although under the framework of a system with one less machine). Specifically, we have

$$\begin{aligned}
\underline{q}_m &= (\underline{q}_{m,0}, \underline{q}_{m,1}, \ldots, \underline{q}_{m,C-1-m}), \\
\underline{q}_{m,0} &= (q_{m,0,1,1}, \ldots, q_{m,0,1,s_1}, q_{m,0,2,1}, \ldots, q_{m,0,2,b_1}), \\
\underline{q}_{m,n} &= (q_{m,n,1,1}, \ldots, q_{m,n,1,s_1}, q_{m,n,2,1}, \ldots, q_{m,n,2,b_1}, q_{m,n,3,1}, \ldots, q_{m,n,3,s_2}, q_{m,n,4,1}, \ldots, q_{m,n,4,b_2}), \\
\underline{q} &= (\underline{q}_{C-1}, \underline{q}_{C-2}, \ldots, \underline{q}_1, \underline{q}_0).
\end{aligned}$$

Note that $\underline{q}$ is a row vector of length

$$\ell = s_1 + (C-1)(s_2+b_2) + \sum_{m=1}^{C-1} [s_1+b_1+(C-1-m)(s_1+b_1+s_2+b_2)] = s_1 + \frac{C(C-1)}{2}(s_1+b_1+s_2+b_2),$$

and $\underline{q}\,\underline{e}' = 1 - q_{0,0,\bullet,\bullet}\gamma_{01}^{[0]} - q_{0,+,3,\bullet}\gamma_{21}^{[0]}$, where $q_{0,0,\bullet,\bullet}\gamma_{01}^{[0]} + q_{0,+,3,\bullet}\gamma_{21}^{[0]}$ is the probability that the machine immediately begins service after suffering a class-1 failure.

If we simply consider how the queue length ahead of the target class-1 customer changes, we can define, for a system with $D$ total machines,

$$\tilde{Q}^{[D]} = \begin{array}{c|ccccccc} & D & D-1 & D-2 & \cdots & 2 & 1 & 0 \\ \hline D & Q^{[D]}_{D,D} & Q^{[D]}_{D,D-1} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ D-1 & \mathbf{0} & Q^{[D]}_{D-1,D-1} & Q^{[D]}_{D-1,D-2} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ D-2 & \mathbf{0} & \mathbf{0} & Q^{[D]}_{D-2,D-2} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ 2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q^{[D]}_{2,2} & Q^{[D]}_{2,1} & \mathbf{0} \\ 1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & Q^{[D]}_{1,1} & \tilde{Q}^{[D]}_{1,0} \\ 0 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \tilde{Q}^{[D]}_{0,0} \end{array},$$

which can serve as the rate matrix for a phase-type representation of the target customer's waiting time distribution, where the level of the process decreases until it is eventually absorbed out of level 1 or level 0. Note that we have retained the contributions from class-1 arrivals on the main diagonal terms of $Q^{[D]}_{m,m}$ and $\tilde{Q}^{[D]}_{0,0}$, as they are ultimately required for the final analysis. For the immediate discussion, however, we proceed as if these were not included, and hence would not cause incidental non-zero row sums that would imply positive transition rates to absorption from unintended states. Moreover, the level of this rate matrix corresponds to the length of the queue in front of the target customer, which is clearly different than the total class-1 queue length. To adjust for this change relative to the original QBD process, and to the fact that the waiting time ends when the target customer is eligible to receive service, we make use of the modified blocks $\tilde{Q}^{[D]}_{1,0}$ and $\tilde{Q}^{[D]}_{0,0}$. Specifically,

$$\tilde{Q}^{[D]}_{0,0} = \begin{array}{c|cccccc} & 0 & 1 & 2 & \cdots & D-1 & D \\ \hline 0 & -D\delta_{\mathcal{I},-2}\alpha I_{s_1} + S_1 & D\delta_{\mathcal{I},-2}\alpha_2\underline{e}'\begin{pmatrix} \underline{\gamma}_{12} & \gamma^{[0]}_{12}\underline{\beta}_2 \end{pmatrix} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ 1 & \begin{bmatrix} \mathbf{0} \\ \underline{B}'_{0,2}\underline{\gamma}_{21} \end{bmatrix} & \Delta^{[D]}_1 & (D-1)\alpha_2 I_{s_2+b_2} & \ddots & \mathbf{0} & \mathbf{0} \\ 2 & \delta_{\mathcal{I},-1}\begin{bmatrix} \mathbf{0} \\ \underline{B}'_{0,2}\underline{\gamma}_{21} \end{bmatrix} & (1-\delta_{\mathcal{I},-1})\Gamma & \Delta^{[D]}_2 & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ D-1 & \delta_{\mathcal{I},-1}\begin{bmatrix} \mathbf{0} \\ \underline{B}'_{0,2}\underline{\gamma}_{21} \end{bmatrix} & \mathbf{0} & \mathbf{0} & \cdots & \Delta^{[D]}_{D-1} & \alpha_2 I_{s_2+b_2} \\ D & \delta_{\mathcal{I},-1}\begin{bmatrix} \mathbf{0} \\ \underline{B}'_{0,2}\underline{\gamma}_{21} \end{bmatrix} & \mathbf{0} & \mathbf{0} & \cdots & (1-\delta_{\mathcal{I},-1})\Gamma & \Delta^{[D]}_D \end{array}$$

is structurally similar to $Q^{[D]}_{0,0}$, with the idle server state and class-0 switch-in states replaced with class-1 switch-in states which lead to absorption. Conditional on $\mathcal{I} = -1$, the transitions after a class-2 service completion are redirected towards these states. To achieve this, we multiply $(1 - \delta_{\mathcal{I},-1})$ into $\Gamma$ to remove those possible transitions, and redirect the system to $X_2(t) = 0$ with the transitions in column 0 of $\tilde{Q}^{[D]}_{0,0}$. If class 1 has non-preemptive priority, then the server will switch to serve class 1 after a service completion, and from the target class-1 customer's perspective, the class-2 queue length no longer matters. For this reason, we also multiply $\delta_{\mathcal{I},-2}$ into the failure rates of other machines, since once they have reached the front of their queue (and the server is switching to serve them), an arrival can only impact the target customer if it is a class-2 failure and class 2 has non-preemptive priority. This would result in the server leaving the target class-1 customer until the class-2 queue empties again. In addition, if the system would transition to these states following a class-2 service completion (with probability $\gamma_{21}^{[0]}$), then the process is directly absorbed without visiting the class-1 switch-in states.

Next, we have

$$\tilde{Q}_{1,0}^{[D]} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ D-2 \\ D-1 \end{array} \!\! \begin{array}{c} \begin{array}{ccccccc} 0 & 1 & 2 & \cdots & D-2 & D-1 & D \end{array} \\ \left[\begin{array}{ccccccc} \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \tilde{Q}_{1,0}^{\star} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \tilde{Q}_{1,0}^{\star} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \tilde{Q}_{1,0}^{\star} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \tilde{Q}_{1,0}^{\star} & \mathbf{0} \end{array}\right] \end{array},$$

where

$$\tilde{Q}_{1,0}^{\star} = \delta_{\mathcal{I},-2} \left[\begin{array}{cc} \mathbf{0} & \mathbf{0} \\ \underline{B}'_{0,1}\underline{\gamma}_{12} & \underline{B}'_{0,1}\gamma_{12}^{[0]}\underline{\beta}_2 \\ \mathbf{0} & \mathbf{0} \end{array}\right] = \delta_{\mathcal{I},-2} Q_{1,0}^{\star}.$$

The definitions of $\tilde{Q}_{1,0}^{[D]}$ and $Q_{1,0}^{[D]}$ are almost identical, except that the block $\tilde{Q}_{1,0}^{[D]}$ leads the process to absorption automatically (instead of visiting level 0 of the process) when $X_2(t) = 0$ or when $X_2(t) \geq 1$ and $\mathcal{I} \neq -2$, as there are no longer any customers ahead of the target customer and the server is already at the class-1 queue.

If the assumption that no class-1 customers could arrive behind the target customer held true, then we could claim that the waiting time is phase-type distributed with representation $\text{PH}_\ell(\underline{q}, \tilde{Q}^{[C-1]})$, as there are $C-1$ customers in the system which are not the target customer (and, in theory, could be queued ahead of it), and during this entire waiting time period, the target customer will never be at risk of failing again. However, this would obviously be an incorrect assumption to make since if a machine experiences a class-1 failure, while it does not add to the list of machines obtaining service ahead of the target customer, it does impact the rate of machines experiencing class-2 failures (due to the finite population assumption) which, depending on the service policy, may need to be serviced before the target customer. To address this issue, we propose the rate matrix

$$\mathcal{R} = \begin{array}{c} \\ C-1 \\ C-2 \\ C-3 \\ \vdots \\ 2 \\ 1 \\ 0 \end{array} \!\! \begin{array}{c} \begin{array}{ccccccc} C-1 & C-2 & C-3 & \cdots & 2 & 1 & 0 \end{array} \\ \left[\begin{array}{ccccccc} \tilde{Q}^{[C-1]} & \tilde{Q}_{-}^{[C-1]} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \tilde{Q}^{[C-2]} & \tilde{Q}_{-}^{[C-2]} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \tilde{Q}^{[C-3]} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \tilde{Q}^{[2]} & \tilde{Q}_{-}^{[2]} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \tilde{Q}^{[1]} & \tilde{Q}_{-}^{[1]} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \tilde{Q}^{[0]} \end{array}\right] \end{array},$$

where

$$\tilde{Q}_{-}^{[D]} = \begin{array}{c} \\ D \\ D-1 \\ D-2 \\ \vdots \\ 2 \\ 1 \\ 0 \end{array} \begin{array}{c} \begin{array}{cccccc} D-1 & D-2 & \cdots & 2 & 1 & 0 \end{array} \\ \left[\begin{array}{cccccc} \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ Q_{D-1,D}^{[D]} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & Q_{D-2,D-1}^{[D]} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & Q_{2,3}^{[D]} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & Q_{1,2}^{[D]} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \tilde{Q}_{0,1}^{[D]} \end{array}\right] \end{array},$$

$$\tilde{Q}_{0,1}^{[D]} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ D-1 \\ D \end{array} \begin{array}{c} \begin{array}{ccccc} 0 & 1 & 2 & \cdots & D-1 \end{array} \\ \left[\begin{array}{ccccc} D\delta_{\mathcal{I},-2}\alpha_1 I_{s_1} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \tilde{Q}_{0,1,1}^{[D]} & \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \tilde{Q}_{0,1,2}^{[D]} & \ddots & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \tilde{Q}_{0,1,D-1}^{[D]} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \end{array}\right] \end{array},$$

and

$$\tilde{Q}_{0,1,j}^{[D]} = (D-j)\alpha_1 I_{s_2+b_2}.$$

Note that through the use of $\tilde{Q}_{-}^{[D]}$, the rate matrix $\mathcal{R}$ can reduce the system size by a single customer whenever a class-1 arrival would be observed. The blocks of $\tilde{Q}_{-}^{[D]}$ include the same $Q_{i,i+1}^{[D]}$ blocks defined previously, as well as a modified $\tilde{Q}_{0,1}^{[D]}$. When the queue length ahead of the target customer is zero, a class-1 arrival no longer increases the range of combinations of $L(t)$ and $Y(t)$ that the system must track from $s_2 + b_2$ to $s_1 + b_2 + s_2 + b_2$.

To pair with the rate matrix $\mathcal{R}$, we define $\underline{\Phi} = (\underline{q}, \underline{0}, \underline{0}, \ldots, \underline{0})$ to be the corresponding initial probability vector of length

$$\begin{aligned} \ell^* &= s_1 + \sum_{i=1}^{C-1} \left( s_1 + \frac{i(i+1)}{2}(s_1 + b_1 + s_2 + b_2) \right) \\ &= Cs_1 + \frac{1}{2}(s_1 + b_1 + s_2 + b_2) \left( \frac{C(C-1)}{2} + \frac{C(C-1)(2(C-1)+1)}{6} \right) \\ &= Cs_1 + \frac{C(C-1)}{4}(s_1 + b_1 + s_2 + b_2) \left( 1 + \frac{1}{3}(2C-1) \right). \end{aligned}$$

The interpretation of $\underline{\Phi}$ is that the arrival of the target customer will always initiate the system in consideration of $C-1$ total other customers, which is only reduced further by future class-1 arrivals. As a result, the waiting time of our target class-1 customer is phase-type distributed with representation $\text{PH}_{\ell^*}(\underline{\Phi}, \mathcal{R})$. Moreover, under exhaustive and non-preemptive priority service policies, a customer's service may not be interrupted, implying that the sojourn time is simply the sum of the waiting time and (independent) service time. Thus, it immediately follows that the class-1 sojourn time is phase-type distributed with representation $\mathrm{PH}_{\ell^*+b_1}((\underline{\Phi}, (q_{0,0,\bullet,\bullet}\gamma_{01}^{[0]} + q_{0,+,3,\bullet}\gamma_{21}^{[0]})\underline{\beta}_1), \mathcal{T})$, where

$$\mathcal{T} = \begin{bmatrix} \mathcal{R} & (-\mathcal{R}\underline{e}')\underline{\beta}_1 \\ \mathbf{0} & B_1 \end{bmatrix}.$$

Moments of the class-1 sojourn time distribution can easily be computed by applying Equation (1.14). Finally, we remark that in order to obtain the corresponding sojourn time distribution for a machine that suffers a class-2 failure, one can simply switch all class-1 and class-2 parameters and distributions (the value of $\mathcal{I}$ will also need to be adjusted if the non-preemptive priority service policy is in place), recalculate the steady-state probabilities, and then repeat the above analysis.

## 2.4 Preemptive Resume Priority Models

We now turn our attention to the preemptive resume priority service policy. The primary way that preemptive resume priority differs from non-preemptive priority is that the arrival of a high priority customer to an empty queue (of their class) will trigger the server to begin a switch-in, independent of their current location. More precisely, the server is now able to interrupt the service of a low priority customer, whereas previously the server would only immediately change location (after observing an arrival) if they were idle or in the midst of a switch-in time. Eventually, once the high priority queue has been emptied again, the server resumes service with the interrupted customer in the low priority queue.

Unlike the previous subsection, whether class 1 or class 2 has preemptive resume priority will greatly impact the derivations needed to characterize the class-1 sojourn time distribution. As such, we consider each case separately in the following two sub-subsections. In Section 2.4.1, we assume that class 1 has preemptive resume priority over class 2 and we determine the distribution of the time spent waiting and in service for a target class-1 customer. In Section 2.4.2, however, class 2 is assumed to have preemptive resume priority over class 1, and we seek to derive the sojourn time distribution of a target class-1 customer.

### 2.4.1 Case 1: $\mathcal{I} = 1$

To model a system in which class 1 has preemptive resume priority, we use the CTMC

$$\{(X_1(t), X_2(t), L(t), Y(t), Y_2(t)), t \geq 0\},$$

where $X_1(t)$, $X_2(t)$, and $L(t)$ are as previously defined in Section 2.3. Moreover, $Y(t)$ denotes the phase of the service (if serving class 1) or switch-in time with possible values depending on $L(t)$ as follows:

$$Y(t) \in \Omega_Y^{[1]}(L(t)) = \begin{cases} \{0\} & \text{, if } L(t) = 0, \\ \{1, 2 \ldots, s_1\} & \text{, if } L(t) = 1, \\ \{1, 2 \ldots, b_1\} & \text{, if } L(t) = 2, \\ \{1, 2 \ldots, s_2\} & \text{, if } L(t) = 3, \\ \{0\} & \text{, if } L(t) = 4, \\ \{1, 2 \ldots, s_0\} & \text{, if } L(t) = 5. \end{cases}$$

The new variable $Y_2(t)$ is intended to keep track of the phase of service of a preempted class-2 customer, taking on values (which depend on $X_2(t)$) according to

$$Y_2(t) \in \Omega_{Y_2}^{[1]}(X_2(t)) = \begin{cases} \{0\} & \text{, if } X_2(t) = 0, \\ \{1, 2, \ldots, b_2\} & \text{, if } X_2(t) \geq 1. \end{cases}$$

Let $\pi_{m,n,l,y,y_2}^{[1]}$ be the steady-state probability of observing the CTMC in state $(m, n, l, y, y_2)$, where $0 \leq m \leq C$, $0 \leq n \leq C - m$, and $l$, $y$, and $y_2$ take values from the respective supports of $L(t)$, $Y(t)$, and $Y_2(t)$, above. With $X_1$ as the level of the process, we define

$$\underline{\pi}_0^{[1]} = (\pi_{0,0,0,0,0}^{[1]}, \pi_{0,0,5,1,0}^{[1]}, \ldots, \pi_{0,0,5,s_0,0}^{[1]}, \underline{\pi}_{0,1}^{[1]}, \ldots, \underline{\pi}_{0,C}^{[1]})$$

to be the ordered steady-state probability row vector for level 0, in which

$$\underline{\pi}_{0,n}^{[1]} = (\pi_{0,n,3,1,1}^{[1]}, \ldots, \pi_{0,n,3,1,b_2}^{[1]}, \pi_{0,n,3,2,1}^{[1]}, \ldots, \pi_{0,n,3,s_2,b_2}^{[1]}, \pi_{0,n,4,0,1}^{[1]}, \ldots, \pi_{0,n,4,0,b_2}^{[1]})$$

is a row vector of length $s_2 b_2 + b_2$ for $n = 1, 2, \ldots, C$. Therefore, level 0 consists of $1 + s_0 + C(s_2 b_2 + b_2)$ total states. For $m = 1, 2, \ldots, C$, the $m^{\text{th}}$ steady-state probability row vector is

$$\underline{\pi}_m^{[1]} = (\underline{\pi}_{m,0}^{[1]}, \underline{\pi}_{m,1}^{[1]}, \ldots, \underline{\pi}_{m,C-m}^{[1]}),$$

where

$$\underline{\pi}_{m,0}^{[1]} = (\pi_{m,0,1,1,0}^{[1]}, \ldots, \pi_{m,0,1,s_1,0}^{[1]}, \pi_{m,0,2,1,0}^{[1]}, \ldots, \pi_{m,0,2,b_1,0}^{[1]}),$$

and for $n = 1, 2, \ldots, C - m$,

$$\begin{aligned} \underline{\pi}_{m,n}^{[1]} = (&\pi_{m,n,1,1,1}^{[1]}, \ldots, \pi_{m,n,1,1,b_2}^{[1]}, \pi_{m,n,1,2,1}^{[1]}, \ldots, \pi_{m,n,1,s_1,b_2}^{[1]}, \\ &\pi_{m,n,2,1,1}^{[1]}, \ldots, \pi_{m,n,2,1,b_2}^{[1]}, \pi_{m,n,2,2,1}^{[1]}, \ldots, \pi_{m,n,2,b_1,b_2}^{[1]}), \end{aligned}$$

which have respective lengths of $s_1 + b_1$ and $(s_1 + b_1)b_2$. Clearly, level $m$ possesses $s_1 + b_1 + (C - m)(s_1 + b_1)b_2$ states for $m \geq 1$. Let $\underline{\pi}^{[1]} = (\underline{\pi}_0^{[1]}, \underline{\pi}_1^{[1]}, \ldots, \underline{\pi}_C^{[1]})$ be the steady-state probability row vector for the full process. For notational convenience, let $Q^{[C,1]}$ now denote the corresponding infinitesimal generator for a system with $C$ machines and class-1 preemptive priority, constructed in the manner of Equation (2.1), but with blocks denoted by $Q_{i,j}^{[C,1]}$ rather than $Q_{i,j}^{[C]}$. Letting $\otimes$ denote the *Kronecker product operator*, the diagonal blocks of $Q^{[C,1]}$ can be expressed as

$$Q_{0,0}^{[C,1]} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C-1 \\ C \end{array} \begin{array}{c} \begin{array}{cccccc} 0 & 1 & 2 & \cdots & C-1 & C \end{array} \\ \left[ \begin{array}{cccccc} \Delta_0^{[C,1]} & C\alpha_2 \underline{e}' \begin{pmatrix} \underline{\gamma}_{02} \otimes \underline{\beta}_2 & \gamma_{02}^{[0]} \underline{\beta}_2 \end{pmatrix} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \begin{bmatrix} \underline{0}' & \mathbf{0} \\ \gamma_{20}^{[0]} \underline{B}_{0,2}' & \underline{B}_{0,2}' \underline{\gamma}_{20} \end{bmatrix} & \Delta_1^{[C,1]} & (C-1)\alpha_2 I_{s_2 b_2 + b_2} & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \Gamma^{[1]} & \Delta_2^{[C,1]} & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \Delta_{C-1}^{[C,1]} & \alpha_2 I_{s_2 b_2 + b_2} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \Gamma^{[1]} & \Delta_C^{[C,1]} \end{array} \right] \end{array},$$

where

$$\Delta_n^{[C,1]} = \begin{cases} -C\alpha I_{1+s_0} + \begin{bmatrix} 0 & \underline{0} \\ \underline{S}'_{0,0} & S_0 \end{bmatrix} & , \text{ if } n = 0, \\ -(C-n)\alpha I_{s_2 b_2 + b_2} + \begin{bmatrix} S_2 \otimes I_{b_2} & \underline{S}'_{0,2} \otimes I_{b_2} \\ \mathbf{0} & B_2 \end{bmatrix} & , \text{ if } n = 1, 2, \ldots, C, \end{cases}$$

and

$$\Gamma^{[1]} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \underline{B}'_{0,2}\underline{\beta}_2 \end{bmatrix},$$

while for $m = 1, 2, \ldots, C$,

$$Q_{m,m}^{[C,1]} = \begin{array}{c|cccccc|} & 0 & 1 & 2 & \cdots & C-m-1 & C-m \\ 0 & Q_{m,m,0}^{[C,1]} & (UD)_{m,0}^{[C,1]} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ 1 & \mathbf{0} & Q_{m,m,1}^{[C,1]} & (UD)_{m,1}^{[C,1]} & \ddots & \mathbf{0} & \mathbf{0} \\ 2 & \mathbf{0} & \mathbf{0} & Q_{m,m,2}^{[C,1]} & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ C-m-1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q_{m,m,C-m-1}^{[C,1]} & (UD)_{m,C-m-1}^{[C,1]} \\ C-m & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & Q_{m,m,C-m}^{[C,1]} \end{array},$$

where

$$Q_{m,m,n}^{[C,1]} = \begin{cases} -(C-m)\alpha I_{s_1+b_1} + \begin{bmatrix} S_1 & \underline{S}'_{0,1}\underline{\beta}_1 \\ \mathbf{0} & B_1 \end{bmatrix} & , \text{ if } n = 0, \\ \\ -(C-m-n)\alpha I_{(s_1+b_1)b_2} + \begin{bmatrix} S_1 & \underline{S}'_{0,1}\underline{\beta}_1 \\ \mathbf{0} & B_1 \end{bmatrix} \otimes I_{b_2} & , \text{ if } n = 1, 2, \ldots, C-m, \end{cases}$$

and

$$(UD)_{m,n}^{[C,1]} = \begin{cases} (C-m)\alpha_2 I_{(s_1+b_1)} \otimes \underline{\beta}_2 & , \text{ if } n = 0, \\ \\ (C-m-n)\alpha_2 I_{(s_1+b_1)b_2} & , \text{ if } n = 1, 2, \ldots, C-m-1. \end{cases}$$

Moving to the off-diagonal blocks of $Q^{[C,1]}$, we first have

$$Q_{m,m+1}^{[C,1]} = \begin{array}{c|ccccc|} & 0 & 1 & 2 & \cdots & C-m-1 \\ 0 & (C-m)\alpha_1 I_{s_1+b_1} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ 1 & \mathbf{0} & Q_{m,m+1,1}^{[C,1]} & \mathbf{0} & \ddots & \mathbf{0} \\ 2 & \mathbf{0} & \mathbf{0} & Q_{m,m+1,2}^{[C,1]} & \ddots & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ C-m-1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q_{m,m+1,C-m-1}^{[C,1]} \\ C-m & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \end{array}$$

for $m = 1, 2, \ldots, C-1$, where

$$Q_{m,m+1,n}^{[C,1]} = (C-m-n)\alpha_1 I_{(s_1+b_1)b_2},\ n = 1, 2, \ldots, C-m-1.$$

Furthermore,

$$Q_{0,1}^{[C,1]} = \begin{array}{c|ccccc|} & 0 & 1 & 2 & \cdots & C-1 \\ 0 & C\alpha_1\underline{e}'\begin{pmatrix} \underline{\gamma}_{01} & \gamma_{01}^{[0]}\underline{\beta}_1 \end{pmatrix} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ 1 & \mathbf{0} & Q_{0,1,1}^{[C,1]} & \mathbf{0} & \ddots & \mathbf{0} \\ 2 & \mathbf{0} & \mathbf{0} & Q_{0,1,2}^{[C,1]} & \ddots & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ C-1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q_{0,1,C-1}^{[C,1]} \\ C & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \end{array},$$

where

$$Q_{0,1,n}^{[C,1]} = (C-n)\alpha_1\underline{e}'\begin{pmatrix} \underline{\gamma}_{21} & \gamma_{21}^{[0]}\underline{\beta}_1 \end{pmatrix} \otimes I_{b_2},\ n = 1, 2, \ldots, C-1,$$

and

$$Q_{1,0}^{[C,1]} = \begin{array}{c|ccccccc|} & 0 & 1 & 2 & \cdots & C-2 & C-1 & C \\ 0 & \begin{bmatrix} \underline{0}' & \mathbf{0} \\ \gamma_{10}^{[0]}\underline{B}'_{0,1} & \underline{B}'_{0,1}\underline{\gamma}_{10} \end{bmatrix} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 1 & \mathbf{0} & Q_{1,0}^{\star,[1]} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 2 & \mathbf{0} & \mathbf{0} & Q_{1,0}^{\star,[1]} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ C-2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q_{1,0}^{\star,[1]} & \mathbf{0} & \mathbf{0} \\ C-1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & Q_{1,0}^{\star,[1]} & \mathbf{0} \end{array},$$

where

$$Q_{1,0}^{\star,[1]} = \begin{bmatrix} \mathbf{0} & \underline{0}' \\ \underline{B}'_{0,1}\underline{\gamma}_{12} & \gamma_{12}^{[0]}\underline{B}'_{0,1} \end{bmatrix} \otimes I_{b_2}.$$

Finally, for $m = 2, 3, \ldots, C$, the remaining blocks of $Q^{[C,1]}$ are given by

$$Q_{m,m-1}^{[C,1]} = \begin{array}{c|ccccccc|} & 0 & 1 & 2 & \cdots & C-m-1 & C-m & C-m+1 \\ 0 & Q_{m,m-1,0}^{[C,1]} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 1 & \mathbf{0} & Q_{m,m-1,1}^{[C,1]} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 2 & \mathbf{0} & \mathbf{0} & Q_{m,m-1,2}^{[C,1]} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ C-m-1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q_{m,m-1,C-m-1}^{[C,1]} & \mathbf{0} & \mathbf{0} \\ C-m & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & Q_{m,m-1,C-m}^{[C,1]} & \mathbf{0} \end{array},$$

where

$$Q^{[C,1]}_{m,m-1,n} = \begin{cases} \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \underline{B}'_{0,1}\underline{\beta}_1 \end{bmatrix} & , \text{ if } n = 0, \\ \\ \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \underline{B}'_{0,1}\underline{\beta}_1 \end{bmatrix} \otimes I_{b_2} & , \text{ if } n = 1, 2, \ldots, C-m. \end{cases}$$

When considering the time a machine spends offline after suffering a class-1 failure, we are again able to decompose the failed machine's sojourn time into its waiting time (to reach the server) and time in service, since a class-1 customer will not experience any service preemptions. We also note that unlike the exhaustive and non-preemptive priority service policies, when considering the class-1 waiting time in isolation, we do not need to track the class-2 queue at all. As a result, we can disregard all arrivals following the target class-1 customer and this greatly simplifies the subsequent analysis. We begin by modifying Equation (2.2) to determine the corresponding steady-state distribution of the system immediately prior to a class-1 customer arrival. Letting $S_{m,n,l,y,y_2}$ denote the event that $(X_1(t), X_2(t), L(t), Y(t), Y_2(t)) = (m, n, l, y, y_2)$ at steady state, we have

$$\begin{aligned} q^{[1]}_{m,n,l,y,y_2} &= \lim_{h \to 0} P(S_{m,n,l,y,y_2} | C_{1,h}) \\ &= \frac{(C-m-n)\pi^{[1]}_{m,n,l,y,y_2}}{\sum_{x_1}\sum_{x_2}\sum_{w}\sum_{z}\sum_{z_2}(C-x_1-x_2)\pi^{[1]}_{x_1,x_2,w,z,z_2}} \,. \end{aligned} \tag{2.4}$$

As before, the right-hand side of Equation (2.4) is equal to zero for all $l$, $y$, and $y_2$ when $m + n = C$.

If a class-1 customer arrives to a non-empty queue, then this arrival does not affect the server and the waiting time is simply the time it takes to empty the queue of class-1 customers in front of this new arrival. On the other hand, if the target customer arrives to find an empty class-1 queue, then the corresponding waiting time is simply equal to the switch-in time, which will lead to an initial probability vector dependent on whether $X_2(t) = 0$ or not. Let

$$q^{[1]}_{0,0,\bullet,\bullet,\bullet} = q^{[1]}_{0,0,0,0,0} + \sum_{i=1}^{s_0} q^{[1]}_{0,0,5,i,0}$$

be the probability that the server is either idle or conducting a switch-in to the idle state immediately before the target class-1 customer arrives (since both queues were empty). Furthermore, let

$$q^{[1]}_{0,+,\bullet,\bullet,\bullet} = \sum_{n=1}^{C-1}\sum_{l=3}^{4}\sum_{y \in \Omega^{[1]}_{Y}(l)}\sum_{y_2 \in \Omega^{[1]}_{Y_2}(n)} q^{[1]}_{0,n,l,y,y_2}$$

be the probability that the target customer arrives to an empty class-1 queue, while $X_2(t) \geq 1$. We separate these two events, despite both yielding a waiting time that only consists of a class-1 switch-in, because the initial probability vector may be different in either case. Taking this into consideration, we may now construct the initial probability vectors for the waiting time distribution. Letting the level of the process equal the number of class-1 customers ahead of the target customer, the initial probability vector corresponding to level 0 is given by

$$\underline{q}^{[1]}_0 = q^{[1]}_{0,0,\bullet,\bullet,\bullet}\underline{\gamma}_{01} + q^{[1]}_{0,+,\bullet,\bullet,\bullet}\underline{\gamma}_{21},$$

which, as we can see, simply initializes the switch-in time, which has a phase-type distribution. For non-zero levels, it is possible for the server to be in the midst of a class-1 switch-in or service time. For each combination of $m$, $l$, and $y \in \{1,2\}$, we obtain the desired marginal distributions by summing the probability mass that was spread out over different states that were used to track $X_2(t)$ or $Y_2(t)$, namely

$$q^{[1]}_{m,\bullet,l,y,\bullet} = q^{[1]}_{m,0,l,y,0} + \sum_{n=1}^{C-m-1} \sum_{y_2 \in \Omega^{[1]}_{Y_2}(n)} q^{[1]}_{m,n,l,y,y_2}. \tag{2.5}$$

Equation (2.5) may then be used to construct the initial probability vector corresponding to level $m$, $1 \leq m \leq C-1$:

$$\underline{q}^{[1]}_{m,\bullet} = (q^{[1]}_{m,\bullet,1,1,\bullet}, \ldots, q^{[1]}_{m,\bullet,1,s_1,\bullet}, q^{[1]}_{m,\bullet,2,1,\bullet}, \ldots, q^{[1]}_{m,\bullet,2,b_1,\bullet}).$$

These vectors may then be collected, including the probability vector for level 0, to construct the full initial probability vector for the process:

$$\underline{q}^{[1]} = (\underline{q}^{[1]}_{C-1,\bullet}, \underline{q}^{[1]}_{C-2,\bullet}, \ldots, \underline{q}^{[1]}_{1,\bullet}, \underline{q}^{[1]}_{0}).$$

Note that $\underline{q}^{[1]}$ is a row vector of length $\ell^{[1]} = s_1 + (C-1)(s_1+b_1)$ and $\underline{q}^{[1]}\underline{e}' = 1 - q^{[1]}_{0,0,\bullet,\bullet,\bullet}\gamma^{[0]}_{01} - q^{[1]}_{0,+,\bullet,\bullet,\bullet}\gamma^{[0]}_{21}$, where $q^{[1]}_{0,0,\bullet,\bullet,\bullet}\gamma^{[0]}_{01} + q^{[1]}_{0,+,\bullet,\bullet,\bullet}\gamma^{[0]}_{21}$ is the probability that the machine immediately begins service after suffering a class-1 failure.

We next focus on designing a rate matrix corresponding to this waiting time for a system that may have up to $D$ customers waiting in front of the target class-1 customer. This ultimately results in

$$\tilde{Q}^{[D,1]} = \begin{array}{c} \\ D \\ D-1 \\ D-2 \\ \vdots \\ 2 \\ 1 \\ 0 \end{array} \begin{array}{c} \begin{array}{ccccccc} D & D-1 & D-2 & \cdots & 2 & 1 & 0 \end{array} \\ \left[\begin{array}{ccccccc} \tilde{Q}^{[D,1]}_{D,D} & \tilde{Q}^{[D,1]}_{D,D-1} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \tilde{Q}^{[D,1]}_{D-1,D-1} & \tilde{Q}^{[D,1]}_{D-1,D-2} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \tilde{Q}^{[D,1]}_{D-2,D-2} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \tilde{Q}^{[D,1]}_{2,2} & \tilde{Q}^{[D,1]}_{2,1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \tilde{Q}^{[D,1]}_{1,1} & \tilde{Q}^{[D,1]}_{1,0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \tilde{Q}^{[D,1]}_{0,0} \end{array}\right] \end{array},$$

where

$$\tilde{Q}^{[D,1]}_{0,0} = S_1$$

is the class-1 switch-in time rate matrix,

$$\tilde{Q}^{[D,1]}_{1,0} = \mathbf{0}$$

is a zero matrix on account of the service completion of the lone customer queueing ahead of the target customer leading to absorption,

$$\tilde{Q}^{[D,1]}_{m,m} = \left[\begin{array}{cc} S_1 & \underline{S}'_{0,1}\underline{\beta}_1 \\ \mathbf{0} & B_1 \end{array}\right], \; m = 1, 2, \ldots, D,$$

can track the $s_1$ switch-in time phases, of which a completion leads to the start of a class-1 service, and

$$\tilde{Q}^{[D,1]}_{m,m-1} = \left[ \begin{array}{cc} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \underline{B}'_{0,1}\underline{\beta}_1 \end{array} \right], \ m = 2, 3, \ldots, D,$$

since a class-1 service completion leads directly into the start of another class-1 service. As previously stated, we do not need to consider any arrivals following that of the target customer, since they do not impact the waiting time. Therefore, the rate matrix corresponding to the waiting time for a class-1 customer in a system with $C$ total customers is simply $\mathcal{R}^{[1]} = \tilde{Q}^{[C-1,1]}$, and it subsequently follows that the waiting time of our target class-1 customer is phase-type distributed with representation $\text{PH}_{\ell^{[1]}}(\underline{q}^{[1]}, \mathcal{R}^{[1]})$. Finally, the class-1 sojourn time distribution of a broken machine, consisting of its waiting time plus an independent service time, can readily be represented as $\text{PH}_{\ell^{[1]}+b_1}((\underline{q}^{[1]}, (q^{[1]}_{0,0,\bullet,\bullet,\bullet}\gamma^{[0]}_{01} + q_{0,+,\bullet,\bullet,\bullet}\gamma^{[0]}_{21})\underline{\beta}_1), \mathcal{T}^{[1]})$, where

$$\mathcal{T}^{[1]} = \left[ \begin{array}{cc} \mathcal{R}^{[1]} & (-\mathcal{R}^{[1]}\underline{e}')\underline{\beta}_1 \\ \mathbf{0} & B_1 \end{array} \right].$$

### 2.4.2 Case 2: $\mathcal{I} = 2$

We now consider the situation in which class 2 has preemptive resume priority over class 1. We remark that while we can use the results for $\mathcal{I} = 1$, by swapping the relevant parameters and distributions, to solve for the steady-state probabilities of the process, as well as the time until repair for a machine that suffers a class-2 failure, we would be unable to characterize the sojourn time distribution for a class-1 failed machine. As such, the purpose of this sub-subsection is to act as a compliment to the analysis of the previous sub-subsection, so that the sojourn time distribution for the lower priority class of machine failures may be found when the server is employing a preemptive resume priority service policy.

First of all, the construction of the infinitesimal generator will involve many of the same techniques used previously, however this time tracking the service phase of the next class-1 customer in line (if any). Moreover, due to the preemptive priority of class-2 customers, the process does not need to consider states where the server is conducting a class-1 switch-in or service time whenever there are class-2 customers in the system. Thus, we model the system by the CTMC

$$\{(X_1(t), X_2(t), L(t), Y(t), Y_1(t)), t \geq 0\},$$

where $X_1(t)$, $X_2(t)$, and $L(t)$ are as previously defined, while $Y(t)$ denotes the phase of the service (if serving class 2) or switch-in time with values depending on $L(t)$ in the following way:

$$Y(t) \in \Omega^{[2]}_Y(L(t)) = \begin{cases} \{0\} & \text{, if } L(t) = 0, \\ \{1, 2, \ldots, s_1\} & \text{, if } L(t) = 1, \\ \{0\} & \text{, if } L(t) = 2, \\ \{1, 2, \ldots, s_2\} & \text{, if } L(t) = 3, \\ \{1, 2, \ldots, b_2\} & \text{, if } L(t) = 4, \\ \{1, 2, \ldots, s_0\} & \text{, if } L(t) = 5. \end{cases}$$

The variable $Y_1(t)$ is used to track the phase of service of a class-1 customer and is determined at the arrival instant of a class-1 customer to an empty queue, as well as upon a service completion of a class-1 customer that segues into the next class-1 service time. Thus, the possible values of $Y_1$ are

$$Y_1(t) \in \Omega_{Y_1}^{[2]}(X_1(t)) = \begin{cases} \{0\} & \text{, if } X_1(t) = 0, \\ \{1, 2, \ldots, b_1\} & \text{, if } X_1(t) \geq 1. \end{cases}$$

We define $\pi_{m,n,l,y,y_1}^{[2]}$ to be the steady-state probability of observing the CTMC in state $(m, n, l, y, y_1)$, where $0 \leq m \leq C$, $0 \leq n \leq C - m$, and $l$, $y$, and $y_1$ take values from the respective supports of $L(t)$, $Y(t)$, and $Y_1(t)$, above. Corresponding to the 0th level of the process, let

$$\underline{\pi}_0^{[2]} = (\pi_{0,0,0,0,0}^{[2]}, \pi_{0,0,5,1,0}^{[2]}, \ldots, \pi_{0,0,5,s_0,0}^{[2]}, \underline{\pi}_{0,1}^{[2]}, \ldots, \underline{\pi}_{0,C}^{[2]}),$$

where

$$\underline{\pi}_{0,n}^{[2]} = (\pi_{0,n,3,1,0}^{[2]}, \ldots, \pi_{0,n,3,s_2,0}^{[2]}, \pi_{0,n,4,1,0}^{[2]}, \ldots, \pi_{0,n,4,b_2,0}^{[2]})$$

is a row vector of length $s_2 + b_2$ for $n = 1, 2, \ldots, C$, so that level 0 has $1 + s_0 + C(s_2 + b_2)$ states. For level $m = 1, 2, \ldots, C$, we define

$$\underline{\pi}_m^{[2]} = (\underline{\pi}_{m,0}^{[2]}, \underline{\pi}_{m,1}^{[2]}, \ldots, \underline{\pi}_{m,C-m}^{[2]}),$$

where

$$\underline{\pi}_{m,0}^{[2]} = (\pi_{m,0,1,1,1}^{[2]}, \ldots, \pi_{m,0,1,1,b_1}^{[2]}, \pi_{m,0,1,2,1}^{[2]}, \ldots, \pi_{m,0,1,s_1,b_1}^{[2]}, \pi_{m,0,2,0,1}^{[2]}, \ldots, \pi_{m,0,2,0,b_1}^{[2]}),$$

and for $n = 1, 2, \ldots, C - m$,

$$\begin{aligned} \underline{\pi}_{m,n}^{[2]} = (&\pi_{m,n,3,1,1}^{[2]}, \ldots, \pi_{m,n,3,1,b_1}^{[2]}, \pi_{m,n,3,2,1}^{[2]}, \ldots, \pi_{m,n,3,s_2,b_1}^{[2]}, \\ &\pi_{m,n,4,1,1}^{[2]}, \ldots, \pi_{m,n,4,1,b_1}^{[2]}, \pi_{m,n,4,2,1}^{[2]}, \ldots, \pi_{m,n,4,b_2,b_1}^{[2]}), \end{aligned}$$

which are row vectors of length $s_1 b_1 + b_1$ and $(s_2 + b_2)b_1$, respectively. In keeping with the same notational convention we adopted in Section 2.4.1, we denote the steady-state probability vector for the overall process by $\underline{\pi}^{[2]} = (\underline{\pi}_0^{[2]}, \underline{\pi}_1^{[2]}, \ldots, \underline{\pi}_C^{[2]})$, which may be obtained via the level-dependent QBD procedure outlined in Section 1.2.6 (in which $Q^{[C,2]}$ denotes the infinitesimal generator for a system with $C$ machines and class-2 preemptive resume priority, structured in the style of Equation (2.1), but with blocks $Q_{i,j}^{[C,2]}$). When considering the blocks of $Q^{[C,2]}$, we first remark that $Q_{0,0}^{[C,2]}$ is actually identical to $Q_{0,0}^{[C]}$ from the exhaustive and non-preemptive priority service models. This is because unlike when $\mathcal{I} = 1$, we must now track phases of class-1 service with our fifth state variable $Y_1(t)$, not class-2 service phases. Since $X_1(t) = 0$ in this block, there are no class-1 service phases to keep track of (i.e., $Y_1(t) = 0$ for all states within this block), and the state space of the level 0 block reduces to that of the aforementioned service models. For $m = 1, 2, \ldots, C$, the other diagonal blocks can be expressed as

$$Q_{m,m}^{[C,2]} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C-m-1 \\ C-m \end{array} \begin{array}{c} \begin{array}{cccccc} 0 & 1 & 2 & \cdots & C-m-1 & C-m \end{array} \\ \left[ \begin{array}{cccccc} Q_{m,m,0}^{[C,2]} & (UD)_{m,0}^{[C,2]} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ (LD)_{m,1}^{[C,2]} & Q_{m,m,1}^{[C,2]} & (UD)_{m,1}^{[C,2]} & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (LD)_{m,2}^{[C,2]} & Q_{m,m,2}^{[C,2]} & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q_{m,m,C-m-1}^{[C,2]} & (UD)_{m,C-m-1}^{[C,2]} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & (LD)_{m,C-m}^{[C,2]} & Q_{m,m,C-m}^{[C,2]} \end{array} \right] \end{array},$$

where

$$Q^{[C,2]}_{m,m,n} = \begin{cases} -(C-m)\alpha I_{s_1b_1+b_1} + \begin{bmatrix} S_1 \otimes I_{b_1} & \underline{S}'_{0,1} \otimes I_{b_1} \\ \mathbf{0} & B_1 \end{bmatrix} & \text{, if } n = 0, \\ \\ -(C-m-n)\alpha I_{(s_2+b_2)b_1} + \begin{bmatrix} S_2 & \underline{S}'_{0,2}\underline{\beta}_2 \\ \mathbf{0} & B_2 \end{bmatrix} \otimes I_{b_1} & \text{, if } n = 1, 2, \ldots, C-m, \end{cases}$$

$$(UD)^{[C,2]}_{m,n} = \begin{cases} (C-m)\alpha_2 \underline{e}' \begin{pmatrix} \underline{\gamma}_{12} & \gamma^{[0]}_{12}\underline{\beta}_2 \end{pmatrix} \otimes I_{b_1} & \text{, if } n = 0, \\ \\ (C-m-n)\alpha_2 I_{(s_2+b_2)b_1} & \text{, if } n = 1, 2, \ldots, C-m-1, \end{cases}$$

and

$$(LD)^{[C,2]}_{m,n} = \begin{cases} \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \underline{B}'_{0,2}\underline{\gamma}_{21} \otimes I_{b_1} & \gamma^{[0]}_{21}\underline{B}'_{0,2} \otimes I_{b_1} \end{bmatrix} & \text{, if } n = 1, \\ \\ \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \underline{B}'_{0,2}\underline{\beta}_2 \end{bmatrix} \otimes I_{b_1} & \text{, if } n = 2, 3, \ldots, C-m. \end{cases}$$

As for the off-diagonal blocks, we first have

$$Q^{[C,2]}_{m,m+1} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C-m-1 \\ C-m \end{array} \begin{array}{c} \begin{array}{ccccc} 0 & 1 & 2 & \cdots & C-m-1 \end{array} \\ \begin{bmatrix} (C-m)\alpha_1 I_{s_1b_1+b_1} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & Q^{[C,2]}_{m,m+1,1} & \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & Q^{[C,2]}_{m,m+1,2} & \ddots & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q^{[C,2]}_{m,m+1,C-m-1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \end{bmatrix} \end{array}$$

for $m = 1, 2, \ldots, C-1$, where

$$Q^{[C,2]}_{m,m+1,n} = (C-m-n)\alpha_1 I_{(s_2+b_2)b_1},\ n = 1, 2, \ldots, C-m-1.$$

In addition,

$$Q^{[C,2]}_{0,1} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C-1 \\ C \end{array} \begin{array}{c} \begin{array}{ccccc} 0 & 1 & 2 & \cdots & C-1 \end{array} \\ \begin{bmatrix} C\alpha_1 \underline{e}' \begin{pmatrix} \underline{\gamma}_{01} \otimes \underline{\beta}_1 & \gamma^{[0]}_{01}\underline{\beta}_1 \end{pmatrix} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & Q^{[C,2]}_{0,1,1} & \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & Q^{[C,2]}_{0,1,2} & \ddots & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q^{[C,2]}_{0,1,C-1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \end{bmatrix} \end{array},$$

where

$$Q^{[C,2]}_{0,1,n} = (C-n)\alpha_1 I_{s_2+b_2} \otimes \underline{\beta}_1,\ n = 1, 2, \ldots, C-1,$$

and

$$Q^{[C,2]}_{1,0} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C-2 \\ C-1 \end{array} \begin{array}{c} \begin{array}{ccccccc} 0 & 1 & 2 & \cdots & C-2 & C-1 & C \end{array} \\ \left[ \begin{array}{ccccccc} \left[ \begin{array}{cc} \underline{0}' & \mathbf{0} \\ \gamma^{[0]}_{10} \underline{B}'_{0,1} & \underline{B}'_{0,1} \underline{\gamma}_{10} \end{array} \right] & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{array} \right] \end{array}.$$

Finally, for $m = 2, 3, \ldots, C$, the remaining blocks of $Q^{[C,2]}$ are given by

$$Q^{[C,2]}_{m,m-1} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C-m-1 \\ C-m \end{array} \begin{array}{c} \begin{array}{ccccccc} 0 & 1 & 2 & \cdots & C-m-1 & C-m & C-m+1 \end{array} \\ \left[ \begin{array}{ccccccc} \left[ \begin{array}{cc} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \underline{B}'_{0,1} \underline{\beta}_1 \end{array} \right] & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{array} \right] \end{array}.$$

When considering the time between when a machine suffers a class-1 failure and when it is up and working again in this particular model, we come to realize that, for the first time, we are unable to uncouple the time spent waiting from the time spent in service. This is due to the unique situation that this service policy presents, in that the target customer's service time can potentially be interrupted due to the arrival of a high priority customer. Therefore, instead of only being concerned about the queue in front of the target class-1 customer emptying, we will model the total time it takes for those in front of the target customer, and the target customer itself, to complete service and leave the system.

Analogous to Equation (2.4), we find that the steady-state probabilities of the system immediately prior to a class-1 arrival can be obtained via

$$q^{[2]}_{m,n,l,y,y_1} = \frac{(C-m-n)\pi^{[2]}_{m,n,l,y,y_1}}{\sum_{x_1}\sum_{x_2}\sum_{w}\sum_{z}\sum_{z_1}(C-x_1-x_2)\pi^{[2]}_{x_1,x_2,w,z,z_1}},$$

which also yields a value of zero for all $l$, $y$, and $y_1$ when $m + n = C$. In anticipation of constructing the various probability vectors involved in characterizing the class-1 sojourn time distribution, we first define

$$q^{[2]}_{0,0,\bullet,\bullet,\bullet} = q^{[2]}_{0,0,0,0,0} + \sum_{i=1}^{s_0} q^{[2]}_{0,0,5,i,0} \tag{2.6}$$

to be the probability that a class-1 arrival finds the server idle or switching into the idle state. In addition, we group the other class-1 arrival instant probabilities into the following row vectors:

$$\begin{aligned}
\underline{q}^{[2]}_{0,n} &= (q^{[2]}_{0,n,3,1,0}, \ldots, q^{[2]}_{0,n,3,s_2,0}, q^{[2]}_{0,n,4,1,0}, \ldots, q^{[2]}_{0,n,4,b_2,0}),\\
\underline{q}^{[2]}_{m,0} &= (q^{[2]}_{m,0,1,1,1}, \ldots, q^{[2]}_{m,0,1,1,b_1}, q^{[2]}_{m,0,1,2,1}, \ldots, q^{[2]}_{m,0,1,s_1,b_1}, q^{[2]}_{m,0,2,0,1}, \ldots, q^{[2]}_{m,0,2,0,b_1}),\\
\underline{q}^{[2]}_{m,n} &= (q^{[2]}_{m,n,3,1,1}, \ldots, q^{[2]}_{m,n,3,1,b_1}, q^{[2]}_{m,n,3,2,1}, \ldots, q^{[2]}_{m,n,3,s_2,b_1},\\
&\quad\; q^{[2]}_{m,n,4,1,1}, \ldots, q^{[2]}_{m,n,4,1,b_1}, q^{[2]}_{m,n,4,2,1}, \ldots, q^{[2]}_{m,n,4,b_2,b_1}).
\end{aligned}$$

We note that if the target class-1 customer does not arrive to find an empty class-1 queue, then this arrival has no impact on any of the variables other than $X_1(t)$. Therefore, letting $\underline{p}_{m,n}$ contain the ordered initial probability masses for states where $X_1(t) = m$ and $X_2(t) = n$, we have

$$\underline{p}_{m+1,n} = \underline{q}^{[2]}_{m,n}, \; m = 1, 2, \ldots, C-1.$$

However, if the target class-1 customer does arrive to find no other class-1 customers present (but with $X_2(t) \geq 1$), the characterization is not as straightforward. Even though the server will not be prompted to move, the first arrival of a class-1 customer requires that the system now track their eventual service phase. Therefore, we let

$$\underline{p}_{1,n} = \underline{q}^{[2]}_{0,n} \otimes \underline{\beta}_1, \; n \geq 1.$$

The last possibility for the arriving target customer involves finding the system empty of customers of either class requiring service, which occurs with probability $q^{[2]}_{0,0,\bullet,\bullet,\bullet}$ given by Equation (2.6). This sees the server begin either a class-1 switch-in time (while the system determines the initial service phase of the target customer), or an immediate class-1 service with probability $\gamma^{[0]}_{0,1}$. Therefore, we define

$$\underline{p}_{1,0} = (q^{[2]}_{0,0,\bullet,\bullet,\bullet}\underline{\gamma}_{01} \otimes \underline{\beta}_1, q^{[2]}_{0,0,\bullet,\bullet,\bullet}\gamma^{[0]}_{01}\underline{\beta}_1).$$

With these pieces in place, we can now define the initial probability vector for the $m^{\text{th}}$ level, $m = 1, 2, \ldots, C$, as $\underline{p}_m = (\underline{p}_{m,0}, \underline{p}_{m,1}, \ldots, \underline{p}_{m,C-m})$, from which we can construct the overall initial probability vector

$$\underline{p} = (\underline{p}_C, \underline{p}_{C-1}, \ldots, \underline{p}_1).$$

We note that the levels of this modified process span from 1 to $C$. This is a result of the actual system immediately prior to the arrival requiring $0 \leq X_1(t) \leq C-1$ in order for a class-1 arrival to be observed, and due to the inclusion of the target customer, the level is incremented by 1. We have no interest in a level 0, since the emptying of the class-1 queue signifies the departure of the target customer, and as we will see below, leads to absorption in a particular CTMC. Incidentally, the row vector $\underline{p}$ has length

$$\sum_{m=1}^{C} [s_1 b_1 + b_1 + (C-m)(s_2 + b_2)s_1] = C(s_1 b_1 + b_1) + \frac{C(C-1)}{2}(s_2 + b_2)s_1,$$

and satisfies $\underline{p}\,\underline{e}' = 1$ (since sojourn times are certain to be positive).

As was the case for the exhaustive and non-preemptive priority service models, we must consider future class-1 arrivals behind the target class-1 customer since they will affect the future arrival rates of class-2 customers, who must all finish service before any class-1 customers may be served. For a model with $D$ total machines, in which there were no class-1 arrivals after the target customer, we would simply have the rate matrix

$$\tilde{Q}^{[D,2]} = \begin{array}{c|cccccc|}
 & D & D-1 & D-2 & \cdots & 2 & 1 \\
D & Q^{[D,2]}_{D,D} & Q^{[D,2]}_{D,D-1} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\
D-1 & \mathbf{0} & Q^{[D,2]}_{D-1,D-1} & Q^{[D,2]}_{D-1,D-2} & \ddots & \mathbf{0} & \mathbf{0} \\
D-2 & \mathbf{0} & \mathbf{0} & Q^{[D,2]}_{D-2,D-2} & \ddots & \mathbf{0} & \mathbf{0} \\
\vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\
2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q^{[D,2]}_{2,2} & Q^{[D,2]}_{2,1} \\
1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & Q^{[D,2]}_{1,1}
\end{array} .$$

In this case, the process is absorbed with rates equal to the service completion rates from $Q^{[D,2]}_{1,0}$ when residing in class-1 service states in level 1. This, of course, cannot accurately describe the entire process. We gather the blocks of $Q^{[D,2]}$ which contain transition rates corresponding to increments of $X_1(t)$ and construct

$$\tilde{Q}^{[D,2]}_{-} = \begin{array}{c|cccccc|}
 & D-1 & D-2 & D-3 & \cdots & 2 & 1 \\
D & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\
D-1 & Q^{[D,2]}_{D-1,D} & \mathbf{0} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} \\
D-2 & \mathbf{0} & Q^{[D,2]}_{D-2,D-1} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} \\
\vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\
3 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\
2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q^{[D,2]}_{2,3} & \mathbf{0} \\
1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & Q^{[D,2]}_{1,2}
\end{array} .$$

Together, these matrices allow us to fully describe the process via the rate matrix

$$\mathcal{R}^{[2]} = \begin{array}{c|cccccc|}
 & C & C-1 & C-2 & \cdots & 2 & 1 \\
C & \tilde{Q}^{[C,2]} & \tilde{Q}^{[C,2]}_{-} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\
C-1 & \mathbf{0} & \tilde{Q}^{[C-1,2]} & \tilde{Q}^{[C-1,2]}_{-} & \ddots & \mathbf{0} & \mathbf{0} \\
C-2 & \mathbf{0} & \mathbf{0} & \tilde{Q}^{[C-2,2]} & \ddots & \mathbf{0} & \mathbf{0} \\
\vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\
2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \tilde{Q}^{[2,2]} & \tilde{Q}^{[2,2]}_{-} \\
1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \tilde{Q}^{[1,2]}
\end{array} ,$$

in combination with the (further) modified initial probability vector $\underline{\Phi}^{[2]} = (\underline{p}, \underline{0}, \underline{0}, \ldots, \underline{0})$ of length

$$\begin{aligned}
\ell^{[2]} &= \sum_{i=1}^{C} \left( i(s_1 b_1 + b_1) + \frac{i(i-1)}{2}(s_2 + b_2)s_1 \right) \\
&= \frac{C(C+1)}{2}(s_1 b_1 + b_1) + \frac{C(C-1)}{4}(s_2 + b_2)s_1 \left( 1 + \frac{1}{3}(2C-1) \right),
\end{aligned}$$

constructed as such since the system will always start in consideration of the full inventory of machines. In conclusion, we deduce that the class-1 sojourn time distribution of a broken machine can be represented as $\mathrm{PH}_{\ell^{[2]}}(\underline{\Phi}^{[2]}, \mathcal{R}^{[2]})$.

## 2.5 Numerical Examples

### 2.5.1 Setup

In this section, we investigate the effect that switch-in times have on the optimality of the different service policies, and the sensitivity of the mean number of working machines on various factors, including the total number of machines as well as the choice of phase-type service time distributions. If we let $\mathcal{S}_i \sim \mathrm{PH}(\underline{\Phi}_i, \mathcal{R}_i)$ denote the random sojourn time of a machine that experiences a class-$i$ failure, $i = 1, 2$, where we suppress each class' dependency on the $\mathcal{I}$, and let $\mathcal{S}$ be the sojourn time of an arbitrary failed machine, then since each failure will independently be a class-$i$ failure with probability $\alpha_i/\alpha$, $i = 1, 2$, it follows that the PDF of $\mathcal{S}$ is

$$f_{\mathcal{S}}(t) = \frac{\alpha_1}{\alpha}\underline{\Phi}_1 \exp\{\mathcal{R}_1 t\}\underline{\mathcal{R}}'_{0,1} + \frac{\alpha_2}{\alpha}\underline{\Phi}_2 \exp\{\mathcal{R}_2 t\}\underline{\mathcal{R}}'_{0,2}, \ t > 0,$$

where $\underline{\mathcal{R}}'_{0,i} = -\mathcal{R}_i\underline{e}'$ is the column vector of absorption rates for the class-$i$ sojourn time distribution. Applying Equation (1.14), the $r^{\text{th}}$ moment for $\mathcal{S}$ has formula

$$\mathrm{E}[\mathcal{S}^r] = (-1)^r r! \left(\frac{\alpha_1}{\alpha}\underline{\Phi}_1\mathcal{R}_1^{-r}\underline{e}' + \frac{\alpha_2}{\alpha}\underline{\Phi}_2\mathcal{R}_2^{-r}\underline{e}'\right). \tag{2.7}$$

In particular, this implies that when $r = 1$, the expected sojourn time satisfies

$$\mathrm{E}[\mathcal{S}] = \frac{\alpha_1}{\alpha}\mathrm{E}[\mathcal{S}_1] + \frac{\alpha_2}{\alpha}\mathrm{E}[\mathcal{S}_2]. \tag{2.8}$$

Throughout this section, we will assume that $\alpha_1/\alpha = 0.9$ and $\alpha_2/\alpha = 0.1$, so that the majority of jobs will belong to class 1.

Let $N_{\mathrm{W}}$ denote the number of working machines. It immediately follows that

$$\mathrm{E}[N_{\mathrm{W}}] = \begin{cases} C - \sum_m \sum_n \sum_l \sum_y (m+n)\pi_{m,n,l,y} & , \text{ if } \mathcal{I} \in \{-2,-1,0\}, \\ C - \sum_m \sum_n \sum_l \sum_y \sum_{y_2} (m+n)\pi^{[1]}_{m,n,l,y,y_2} & , \text{ if } \mathcal{I} = 1, \\ C - \sum_m \sum_n \sum_l \sum_y \sum_{y_1} (m+n)\pi^{[2]}_{m,n,l,y,y_1} & , \text{ if } \mathcal{I} = 2. \end{cases} \tag{2.9}$$

In order to gain efficiency from the priority service policies, we assume that the stratification of jobs into two classes is done in a logical manner such that 'small' jobs and 'large' jobs are not grouped together. Without loss of generality, we allow class-1 customers to have smaller service requirements. The biggest disadvantage to using priority service policies is that they result in more frequent switching between queues by the server. When these switches require non-insignificant amounts of time to complete, the additional time spent not serving customers may reduce the overall system efficiency. Therefore, we begin by considering the effect of $p_{>0} = 1 - \gamma^{[0]}_{ji}$, the probability of a switch-in time from queue $j$ to queue $i$ being non-zero.

Let the corresponding initial probability vectors and rate matrices for the phase-type switch-in time distributions be given by

$$\begin{aligned} \underline{\gamma}_{10} &= (p_{>0}, 0), & \underline{\gamma}_{20} &= (0, p_{>0}), \\ \underline{\gamma}_{01} &= (0, p_{>0}, 0), & \underline{\gamma}_{21} &= (p_{>0}, 0, 0), \\ \underline{\gamma}_{02} &= (0, p_{>0}, 0), & \underline{\gamma}_{12} &= (p_{>0}, 0, 0), \end{aligned}$$

and

$$S_0 = \frac{1}{M_\text{S}} \begin{bmatrix} -2 & 0 \\ 0 & -1 \end{bmatrix},\ S_1 = \frac{1}{M_\text{S}} \begin{bmatrix} -1 & 1 & 0 \\ 0 & -2 & 2 \\ 0 & 0 & -2 \end{bmatrix},\ S_2 = \frac{1}{M_\text{S}} \begin{bmatrix} -2 & 2 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & -1 \end{bmatrix},$$

where $M_\text{S}$ is a constant that allows us to scale the expected switch-in times. We may interpret the above as class-dependent Erlang-2 ($\text{E}_2$) set-up and exponential take-down times, with both being faster for class 1. If the server moves to class 0 instead of the opposite queue (due to it being empty), they may complete the take-down for their previous queue and only require a set-up following the next arrival.

For the service times, we consider hyperexponential-2 ($\text{H}_2$) distributions, with initial probability vectors and rate matrices given by

$$\underline{\beta}_1 = \underline{\beta}_2 = (0.9, 0.1),\ B_1 = 2 \begin{bmatrix} -1 & 0 \\ 0 & -\frac{1}{11} \end{bmatrix},\ \text{and}\ B_2 = \frac{1}{10M_\text{B}} \begin{bmatrix} -1 & 0 \\ 0 & -\frac{1}{11} \end{bmatrix},$$

as well as Erlang-3 ($\text{E}_3$) distributions with phase-type components

$$\underline{\beta}_1 = \underline{\beta}_2 = (1, 0, 0),\ B_1 = \begin{bmatrix} -3 & 3 & 0 \\ 0 & -3 & 3 \\ 0 & 0 & -3 \end{bmatrix},\ \text{and}\ B_2 = \frac{1}{20M_\text{B}} \begin{bmatrix} -3 & 3 & 0 \\ 0 & -3 & 3 \\ 0 & 0 & -3 \end{bmatrix},$$

where, in a similar fashion, $M_\text{B}$ is a constant for scaling the mean class-2 service time. The mean class-1 service time is set equal to 1, whereas the mean class-2 service time is set equal to 20 (when $M_\text{B} = 1$).

Through the use of the $\text{H}_2$ distributions, we are, in effect, considering the mixtures of two exponential distributions, representing the grouping of more than one type of failure within each class. The $\text{E}_3$ distributions enable us to represent the possibility of having a partially completed service to return to (since we allow preemptive resume). While both sets of distributions result in equal means for a given $M_\text{B}$, the $\text{E}_3$ distributions have smaller variances.

### 2.5.2 Simultaneous Optimization of $\text{E}[N_\text{W}]$ and $\text{E}[\mathcal{S}]$

Figure 2.2 contains plots of both $\text{E}[\mathcal{S}]$ and $\text{E}[N_\text{W}]$ using $\text{H}_2$ service with $M_\text{B} = 1$, $C = 10$, $\alpha = 0.075$ (i.e., $\alpha_1 = 0.0675$ and $\alpha_2 = 0.0075$), and $M_\text{S} = 1$, while varying $p_{>0} \in [0, 1]$. Rounding to five decimal places, we observe that for $0 \leq p_{>0} < 0.13351$, class-1 preemptive priority (i.e., $\mathcal{I} = 1$) is optimal in terms of minimizing the mean sojourn time and maximizing the mean number of working machines, whereas class-1 non-preemptive priority (i.e., $\mathcal{I} = -1$) is optimal for $0.13351 \leq p_{>0} < 0.68277$, and exhaustive (i.e., $\mathcal{I} = 0$) is optimal otherwise. Based on our earlier intuition concerning switch-in times and priority service policies, this makes sense. It is optimal for the server to switch upon every class-1 failure when the probability of experiencing a non-zero switch-in time is minimal, but as this probability increases, it no longer becomes optimal to interrupt a class-2 service, eventually reaching the point where the server wishes to eliminate any unnecessary switches. An important observation here is that the optimality of $\mathcal{I}$ changes simultaneously for both the mean sojourn time and mean number of working machines.

We are able to make similar conclusions between the effect of switch-in times and priority service policy optimality from Figure 2.3, by setting $p_{>0} = 1$ and letting $M_\text{S}$ range between 0

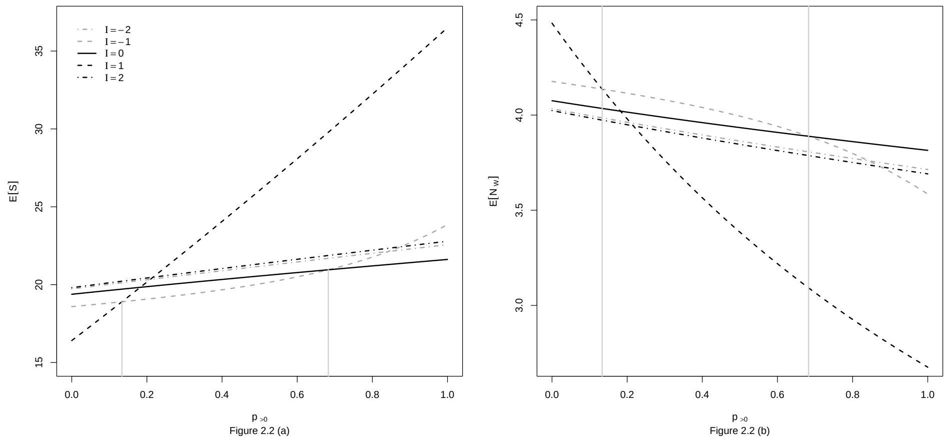

Figure 2.2: Plots of E[$\mathcal{S}$] and E[$N_{\mathrm{W}}$] versus $p_{>0}$ (along with vertical lines indicating values of $p_{>0}$ where the optimal choice of $\mathcal{I}$ changes), with $C = 10$, $\alpha = 0.075$, $M_{\mathrm{S}} = 1$, and $\mathrm{H}_2$ service with $M_{\mathrm{B}} = 1$.

and 2. Even with a guaranteed positive switch-in time, class-1 preemptive priority is optimal for the smallest mean values. This is followed by a small range where class-1 non-preemptive priority is optimal, followed by exhaustive, which continues to be the best choice as $M_{\mathrm{S}}$ becomes large. In both Figures 2.2 and 2.3, we remark at how fast class-1 preemptive priority switches from being the best choice to being the worst, as the cost of the extra incurred switch-in times becomes too large. In these examples, class-1 non-preemptive priority at its worst is not too far from the class-2 priority models in Figure 2.2, but as the mean switch-in times themselves are increasing in Figure 2.3, the total amount of idle time we are 'risking' is increasing and the higher rate of class-1 failures makes class-1 non-preemptive priority vastly under-perform the class-2 priority service policies at large values of $M_{\mathrm{S}}$.

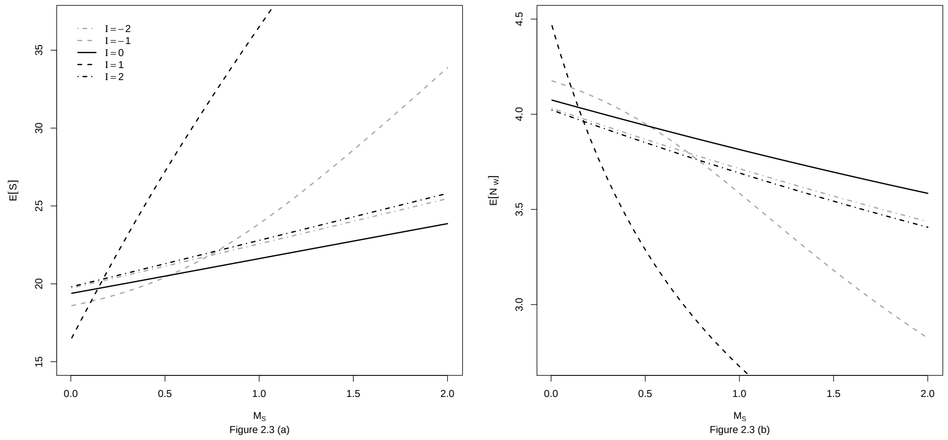

Figure 2.3: Plots of E[$\mathcal{S}$] and E[$N_{\mathrm{W}}$] versus $M_{\mathrm{S}}$, with $C = 10$, $\alpha = 0.075$, $p_{>0} = 1$, and $\mathrm{H}_2$ service with $M_{\mathrm{B}} = 1$.

These observations are not coincidences, as they hold by the following theorem.

**Theorem 2.1.** *For a maintenance system with $C$ machines and a given failure rate $\alpha$,* $\mathrm{E}[N_{\mathrm{W}}]$ *will simultaneously be maximized while* $\mathrm{E}[\mathcal{S}]$ *is minimized.*

*Proof.* Little's Law [64] states that the expected number of customers present in a system is equal to the expected amount of time a customer spends in a system, multiplied by the average arrival rate. For many models, said arrival rate is constant, and corresponds to one or more Poisson processes that are independent of the rest of the system. However, within this model, customers 'arrive' as machines fail at a rate directly proportional to the number of working machines. In that way, the average arrival rate satisfies $\bar{\alpha} = \alpha\mathrm{E}[N_{\mathrm{W}}]$. Treating the length of the class-$i$ queue as a subsystem, the mean arrival rate to that subsystem is the mean class-$i$ failure rate, $\bar{\alpha}_i = \alpha_i\mathrm{E}[N_{\mathrm{W}}]$, and the time spent in the subsystem by a target machine is of course distributed as a class-$i$ sojourn time. Thus, applying Little's Law, we obtain

$$\mathrm{E}[X_i] = \alpha_i\mathrm{E}[N_{\mathrm{W}}]\mathrm{E}[\mathcal{S}_i],\ \ i = 1, 2. \tag{2.10}$$

Summing Equation (2.10) for $i = 1, 2$ and applying Equation (2.8), we observe that

$$\begin{aligned}\mathrm{E}[X_1] + \mathrm{E}[X_2] &= \alpha_1\mathrm{E}[N_{\mathrm{W}}]\mathrm{E}[\mathcal{S}_1] + \alpha_2\mathrm{E}[N_{\mathrm{W}}]\mathrm{E}[\mathcal{S}_2] \\ &= \alpha\mathrm{E}[N_{\mathrm{W}}]\left(\frac{\alpha_1}{\alpha}\mathrm{E}[\mathcal{S}_1] + \frac{\alpha_2}{\alpha}\mathrm{E}[\mathcal{S}_2]\right) \\ &= \alpha\mathrm{E}[N_{\mathrm{W}}]\mathrm{E}[\mathcal{S}], \end{aligned} \tag{2.11}$$

which can also be obtained through Little's Law by treating the total collection of failed machines as a single subsystem.

We may re-express Equation (2.9) as $\mathrm{E}[N_{\mathrm{W}}] = C - \mathrm{E}[X_1] - \mathrm{E}[X_2]$. Subtracting both sides of Equation (2.11) from $C$, using this expression, and isolating for $\mathrm{E}[N_{\mathrm{W}}]$, we find that

$$\mathrm{E}[N_{\mathrm{W}}] = \frac{C}{1 + \alpha\mathrm{E}[S]}.$$

While this is not a linear relationship, it is clear that the selection of a service policy that maximizes $\mathrm{E}[N_{\mathrm{W}}]$ for a given $C$ and $\alpha$ must simultaneously minimize $\mathrm{E}[\mathcal{S}]$.

□

### 2.5.3 Limiting Behaviour

Figures 2.4 and 2.5 plot $\mathrm{E}[N_{\mathrm{W}}]$ versus $C \in \{2, 3, \ldots, 18\}$ using $\mathrm{E}_3$ service with different combinations of $p_{>0} \in \{0, 0.5, 1\}$, $\alpha \in \{0.05, 0.1\}$, $M_{\mathrm{B}} \in \{0.5, 1\}$, and $M_{\mathrm{S}} \in \{1, 2\}$. We observe that as we increase $C$, $\mathrm{E}[N_{\mathrm{W}}]$ converges to some constant value that depends on $\mathcal{I}$ when $p_{>0}$ is positive. This is a consequence of the following theorem, where we let $\mathrm{N}_W^{[C]}$ represent the number of working machines' dependency on $C$. The corresponding limits or upper bounds from Theorem 2.2 are presented within these figures by light grey horizontal lines.

**Theorem 2.2.** *For any service policy, the limit of the number of working machines satisfies*

$$\mathrm{E}[N_{\mathrm{W}}^{[\infty]}] = \lim_{C\to\infty}\mathrm{E}[N_{\mathrm{W}}^{[C]}] \le \frac{-1}{\alpha_1\underline{\beta}_1 B_1^{-1}\underline{e}' + \alpha_2\underline{\beta}_2 B_2^{-1}\underline{e}'}. \tag{2.12}$$

*Additionally, if switch-in times between the class-*1 *and class-*2 *queues are identically zero (i.e.,* $\gamma_{ji}^{[0]} = 1 \ \forall \ i, j \in \{1, 2\}$*), then the upper bound will surely be reached, i.e.,*

$$\mathrm{E}[N_{\mathrm{W}}^{[\infty]}] = \frac{-1}{\alpha_1 \underline{\beta}_1 B_1^{-1} \underline{e}' + \alpha_2 \underline{\beta}_2 B_2^{-1} \underline{e}'}. \tag{2.13}$$

*Proof.* This is a special case of Theorem 3.2, which is proven in the Appendix.

□

For the examples we are considering, this upper bound or limit equals

$$\frac{-1}{\alpha_1 \underline{\beta}_1 B_1^{-1} \underline{e}' + \alpha_2 \underline{\beta}_2 B_2^{-1} \underline{e}'} = \frac{1/\alpha}{(0.9)(1) + (0.1)(20 M_{\mathrm{B}})}. \tag{2.14}$$

The presence of a limit of $\mathrm{E}[N_{\mathrm{W}}]$ as we increase $C$ is a result of the existence of a tipping point where the server's rate of fixing machines balances out with the rate of machine failures. Any further machines introduced into the system after this limit is reached will effectively increase the average number of broken machines by 1. For more details, refer to Remark 3.2.

When $p_{>0} = 0$, the additional switches that a server experiences from a priority service policy do not result in any idle time, and so each policy converges to the same value of $\mathrm{E}[N_{\mathrm{W}}]$, albeit at different rates. When $p_{>0} = 0.5$, we observe that each service policy now converges to a different value of $\mathrm{E}[N_{\mathrm{W}}]$. This is due to the fact that different priority service policies introduce different amounts of extra switches, which result in different percentages of time that the server is idle. The higher percentage of time that the server is idle, the smaller the net rate of repaired machines per unit time. As the probability of a failure coming from class 1 is much higher than that of class 2, class-1 preemptive priority results in the highest amount of extra switch-ins due to the long class-2 service times, followed by class-1 non-preemptive priority. The class-2 priority policies introduce similar amounts of extra switch-ins due to a combination of the lower frequency of class-2 failures and the faster class-1 service times. At $p_{>0} = 1$, this difference is further amplified and we see an increased amount of separation. A consequence of this is that the exhaustive service policy always converges to the highest value of $\mathrm{E}[N_{\mathrm{W}}]$ as $C \to \infty$, as it experiences the minimum number of switches, but as it does not necessarily do so at the fastest rate and other policies may yield a higher $\mathrm{E}[N_{\mathrm{W}}]$ at a particular value of $C$.

We observe Equation (2.14)'s dependency on $\alpha$ by comparing Figures 2.4 (a), (c), and (e), against Figures 2.4 (b), (d), and (f). Clearly, the higher rate of failure causes a reduction in all converged values, given that the server's rate of repair is unchanged. Additionally, increasing $\alpha$ results in a faster rate of occurrence for both failure classes, and the spread of converged values of $\mathrm{E}[N_{\mathrm{W}}]$ for each service policy is wider as the extra amount of idle time is increased. Moreover, this increases the rate of convergence to their limits, as each additional working machine contributes a larger amount to the total rate of failure.

We next compare Figures 2.4 (a), (c), and (e), against Figures 2.5 (a), (c), and (e), to ascertain the impact of increasing $M_{\mathrm{S}}$. Similar to increasing $p_{>0}$, at positive values of $p_{>0}$, we remark that this penalizes the priority service policies proportional to their amount of extra incurred switch-ins. As the exhaustive service policy has minimal incurred switch-ins, its converged $\mathrm{E}[N_{\mathrm{W}}]$ values are impacted the least.

Finally, observing Figures 2.4 (a), (c), and (e), and Figures 2.5 (b), (d), and (f), we note that the ratio of mean service times between the two classes is affected. In Figures 2.5 (b),

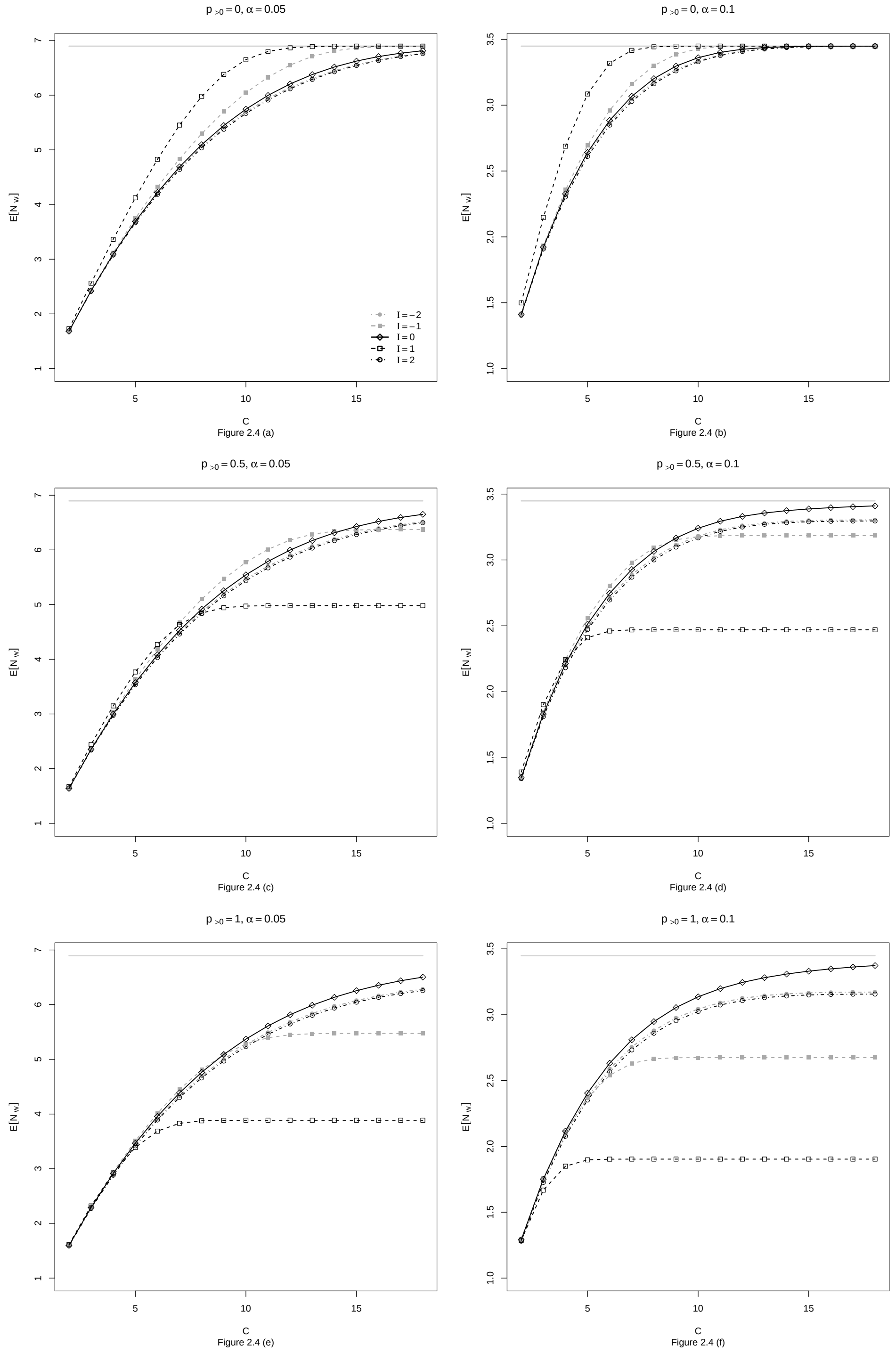

Figure 2.4 (a)

Figure 2.4 (b)

Figure 2.4 (c)

Figure 2.4 (d)

Figure 2.4 (e)

Figure 2.4 (f)

Figure 2.4: Plots of $\mathrm{E}[N_\mathrm{W}]$ versus $C$ under $\mathrm{E}_3$ service and fixed $M_\mathrm{S} = 1$ and $M_\mathrm{B} = 1$, for varying $p_{>0}$ and $\alpha$.

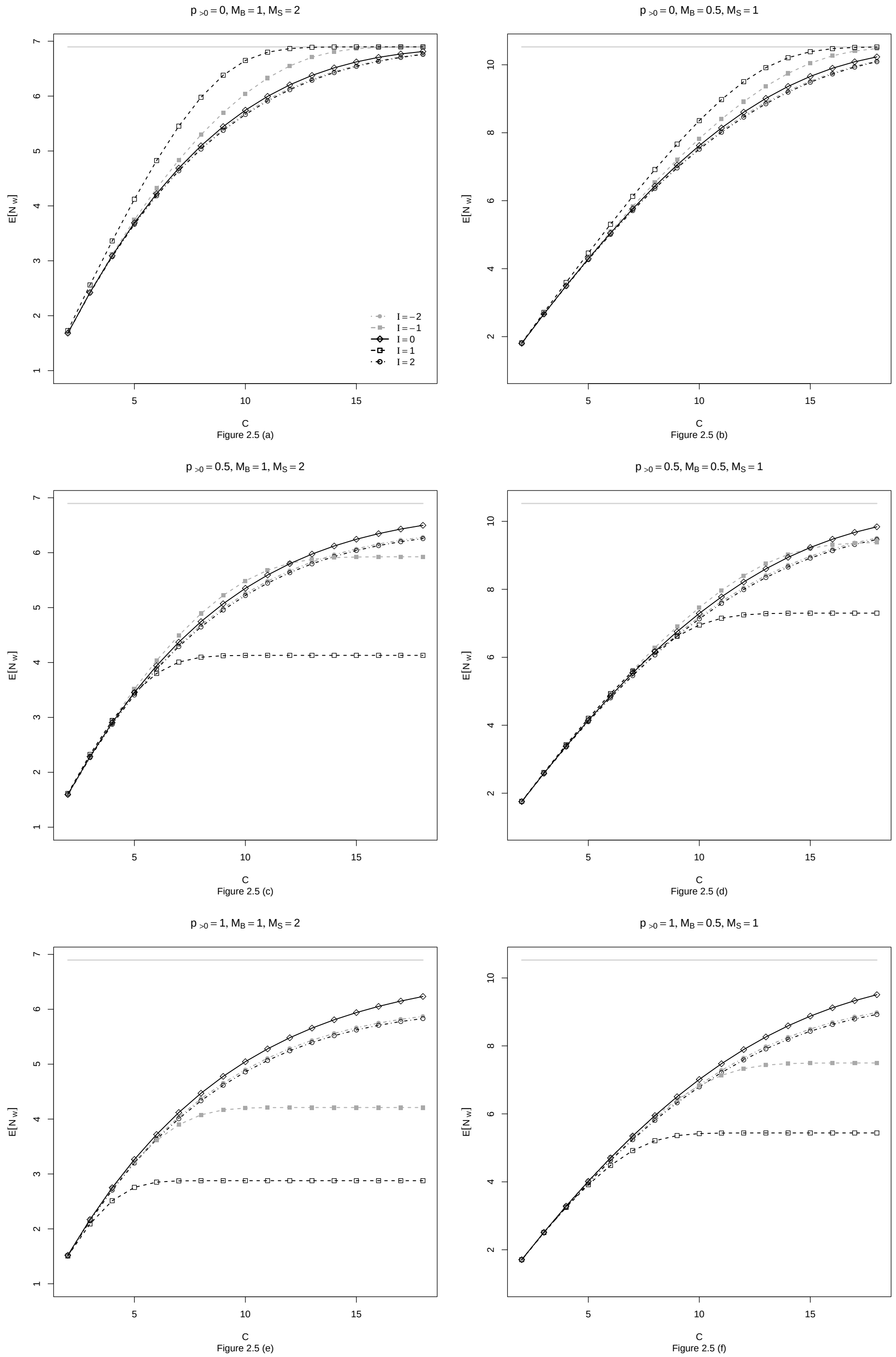

Figure 2.5 (a)

Figure 2.5 (b)

Figure 2.5 (c)

Figure 2.5 (d)

Figure 2.5 (e)

Figure 2.5 (f)

Figure 2.5: Plots of $\mathrm{E}[N_\mathrm{W}]$ versus $C$ under $\mathrm{E}_3$ service and fixed $\alpha = 0.05$, for varying $p_{>0}$, $M_\mathrm{S}$, and $M_\mathrm{B}$.

(d), and (f), we have $M_{\mathrm{B}} = 0.5$, which halves the mean class-2 service time while leaving the class-1 service time distribution unchanged. This increases the rate at which the server repairs machines, and so the rates of convergence are slower to higher final values, as observed in Equation (2.14). The quicker class-2 service times reduce the effectiveness of the class-1 priority policies (while marginally improving the class-2 priority policies), so this narrows the differences in $\mathrm{E}[N_{\mathrm{W}}]$ between the priority service policies and the exhaustive service policy.

### 2.5.4 Optimization Problem

If additional machines were cost-free, then a factory could achieve a maximum expected rate of output production by selecting an exhaustive service policy and increasing $C$ to an arbitrarily large value. However, in the real world, there are in fact restrictions on how many machines can be purchased, either due to capital or space restrictions. Due to the existence of costs, the correct decision may be to use a priority service policy at a value of $C$ that results in a higher value of $\mathrm{E}[N_{\mathrm{W}}]$ than the exhaustive service policy. To approximate this, we introduce the objective function $\mathrm{E}[N_{\mathrm{W}}] - rC$, where $r$ is the cost of possession for each machine in the system. This constant $r$ can be interpreted as the cost per unit time as a fraction of the profit per unit time that a single working machine produces. In this case, the optimal choice of $C$ and $\mathcal{I}$ will maximize our expected profit per unit time. Alternatively, $r$ may be treated as the tolerance that we select to determine if $\mathrm{E}[N_{\mathrm{W}}]$ has converged, such that the objective function will be locally maximized for a given $\mathcal{I}$ at the highest value of $C$ before every additional machine added to the system results in an increase in $\mathrm{E}[N_{\mathrm{W}}]$ of less than $r$ units. Global maximization in this case tells us which service policy converges within the tolerance to the highest value, the fastest.

In Tables 2.1 and 2.2, we provide the optimal $C$ and $\mathcal{I}$ under the $\mathrm{H}_2$ and $\mathrm{E}_3$ service time distributions, respectively, over our previously considered values of $M_{\mathrm{B}}$, $M_{\mathrm{S}}$, $\alpha$, and $p_{>0}$. Additionally, we consider values of the cost parameter $r \in \{0.05, 0.1, 0.25\}$. Comparing these tables, it is clear that the smaller service time variance of the $\mathrm{E}_3$ distributions causes the objective function to converge to higher values, often at smaller values of $C$. Here, a smaller service time variance reduces the probability of the server being stuck on one job for an unusually long period of time, resulting in machines being repaired at a more consistent rate. However, we do not observe a large impact on the optimal choices of $\mathcal{I}$, outside of the case when $M_{\mathrm{B}} = 0.5$ and $M_{\mathrm{S}} = 2$, where the optimal $C$ values for the $\mathrm{E}_3$ distributions are higher. Here, we see that exhaustive service is preferred over class-1 non-preemptive priority, which we would expect to observe at higher values of $C$.

When $p_{>0} = 0$, all service policies converge to the same value of $\mathrm{E}[N_{\mathrm{W}}]$ (all else being equal), but the class-1 preemptive priority policy is universally preferred as it converges at the fastest rate. For moderate values of $p_{>0}$, either class-1 non-preemptive priority or exhaustive service is optimal, largely conditional on $r$, $M_{\mathrm{B}}$, and $M_{\mathrm{S}}$. For larger $r$, the cost per machine is higher, so that the objective function will maximize at a lower value of $C$. As the exhaustive service policy is best for large $C$, but not necessarily small $C$, it is possible for the optimal $C$ to end up in the range where class-1 non-preemptive priority results in a higher value of $\mathrm{E}[N_{\mathrm{W}}]$. Reducing the mean class-2 service time, as observed in Figures 2.4 and 2.5, causes the objective function to maximize at higher values of $C$, to a larger expected profit per unit time. For moderate values of $p_{>0}$, this may result in exhaustive service being preferred over class-1 non-preemptive priority. Finally, as $M_{\mathrm{S}}$ increases, the additional switch-in times that the non-preemptive priority service policy causes reduces the region where $\mathcal{I} = -1$ outperforms

Table 2.1: Optimal combinations of $C$ and $\mathcal{I}$ under $H_2$ service.

| | | | | | | | $p_{>0}$ | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $r = 0.05$ | | | 0 | | | 0.5 | | | 1 | | |
| $M_B$ | $M_S$ | $\alpha$ | $C$ | $\mathcal{I}$ | $E[N_W]$ | $C$ | $\mathcal{I}$ | $E[N_W]$ | $C$ | $\mathcal{I}$ | $E[N_W]$ |
| 1 | 1 | 0.05 | 15 | 1 | 6.8352 | 18 | 0 | 6.4476 | 18 | 0 | 6.3070 |
| | | 0.075 | 11 | 1 | 4.5423 | 15 | 0 | 4.3593 | 15 | 0 | 4.2557 |
| | | 0.10 | 9 | 1 | 3.4025 | 12 | 0 | 3.2409 | 12 | 0 | 3.1517 |
| | 2 | 0.05 | 15 | 1 | 6.8352 | 18 | 0 | 6.2903 | 18 | 0 | 6.0387 |
| | | 0.075 | 11 | 1 | 4.5423 | 16 | 0 | 4.2961 | 17 | 0 | 4.1733 |
| | | 0.10 | 9 | 1 | 3.4025 | 12 | 0 | 3.1421 | 13 | 0 | 3.0430 |
| 0.5 | 1 | 0.05 | 18 | 1 | 10.2860 | 18 | 0 | 9.2835 | 18 | 0 | 9.0040 |
| | | 0.075 | 15 | 1 | 6.9608 | 18 | 0 | 6.6601 | 18 | 0 | 6.4709 |
| | | 0.10 | 12 | 1 | 5.2079 | 16 | 0 | 5.0348 | 17 | 0 | 4.9405 |
| | 2 | 0.05 | 18 | 1 | 10.2860 | 18 | 0 | 8.9543 | 18 | 0 | 8.4692 |
| | | 0.075 | 15 | 1 | 6.9608 | 18 | 0 | 6.4526 | 18 | 0 | 6.1226 |
| | | 0.10 | 12 | 1 | 5.2079 | 17 | 0 | 4.9323 | 18 | 0 | 4.7578 |
| $r = 0.1$ | | | | | | | | | | | |
| 1 | 1 | 0.05 | 13 | 1 | 6.6948 | 15 | -1 | 6.2379 | 17 | 0 | 6.2189 |
| | | 0.075 | 10 | 1 | 4.4827 | 11 | -1 | 4.1039 | 12 | 0 | 4.0405 |
| | | 0.10 | 8 | 1 | 3.3439 | 9 | -1 | 3.0572 | 9 | 0 | 2.9313 |
| | 2 | 0.05 | 13 | 1 | 6.6948 | 17 | 0 | 6.2002 | 17 | 0 | 5.9419 |
| | | 0.075 | 10 | 1 | 4.4827 | 12 | 0 | 4.0240 | 12 | 0 | 3.8182 |
| | | 0.10 | 8 | 1 | 3.3439 | 9 | 0 | 2.9156 | 9 | 0 | 2.7410 |
| 0.5 | 1 | 0.05 | 18 | 1 | 10.2860 | 18 | 0 | 9.2835 | 18 | 0 | 9.0040 |
| | | 0.075 | 13 | 1 | 6.8193 | 16 | 0 | 6.4965 | 16 | 0 | 6.2871 |
| | | 0.10 | 11 | 1 | 5.1484 | 13 | 0 | 4.8391 | 13 | 0 | 4.6614 |
| | 2 | 0.05 | 18 | 1 | 10.2860 | 18 | 0 | 8.9543 | 18 | 0 | 8.4692 |
| | | 0.075 | 13 | 1 | 6.8193 | 17 | 0 | 6.3640 | 18 | 0 | 6.1226 |
| | | 0.10 | 11 | 1 | 5.1484 | 13 | 0 | 4.6380 | 14 | 0 | 4.4440 |
| $r = 0.25$ | | | | | | | | | | | |
| 1 | 1 | 0.05 | 10 | 1 | 6.1367 | 11 | -1 | 5.5854 | 10 | 0 | 5.0791 |
| | | 0.075 | 7 | 1 | 3.9922 | 7 | -1 | 3.4423 | 7 | 0 | 3.2798 |
| | | 0.10 | 6 | 1 | 3.0645 | 5 | -1 | 2.4134 | 5 | 0 | 2.2979 |
| | 2 | 0.05 | 10 | 1 | 6.1367 | 10 | -1 | 5.1427 | 10 | 0 | 4.8054 |
| | | 0.075 | 7 | 1 | 3.9922 | 7 | -1 | 3.3006 | 6 | 0 | 2.8259 |
| | | 0.10 | 6 | 1 | 3.0645 | 5 | -1 | 2.3031 | 4 | 0 | 1.8741 |
| 0.5 | 1 | 0.05 | 15 | 1 | 9.7449 | 15 | -1 | 8.6819 | 16 | 0 | 8.5677 |
| | | 0.075 | 11 | 1 | 6.4953 | 10 | -1 | 5.4877 | 10 | 0 | 5.2170 |
| | | 0.10 | 8 | 1 | 4.6639 | 8 | -1 | 4.0735 | 8 | 0 | 3.8568 |
| | 2 | 0.05 | 15 | 1 | 9.7449 | 16 | 0 | 8.5116 | 16 | 0 | 8.0301 |
| | | 0.075 | 11 | 1 | 6.4953 | 10 | -1 | 5.1787 | 10 | 0 | 4.8051 |
| | | 0.10 | 8 | 1 | 4.6639 | 7 | -1 | 3.5766 | 7 | 0 | 3.2658 |

Table 2.2: Optimal combinations of $C$ and $\mathcal{I}$ under E$_3$ service.

| | | | | | | | $p_{>0}$ | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $r = 0.05$ | | | | 0 | | | 0.5 | | | 1 | |
| $M_B$ | $M_S$ | $\alpha$ | $C$ | $\mathcal{I}$ | E[$N_W$] | $C$ | $\mathcal{I}$ | E[$N_W$] | $C$ | $\mathcal{I}$ | E[$N_W$] |
| 1 | 1 | 0.05 | 12 | 1 | 6.8660 | 18 | 0 | 6.6507 | 18 | 0 | 6.5040 |
| | | 0.075 | 9 | 1 | 4.5771 | 13 | 0 | 4.3870 | 14 | 0 | 4.3240 |
| | | 0.10 | 7 | 1 | 3.4162 | 11 | 0 | 3.2938 | 11 | 0 | 3.1979 |
| | 2 | 0.05 | 12 | 1 | 6.8660 | 18 | 0 | 6.4986 | 18 | 0 | 6.2331 |
| | | 0.075 | 9 | 1 | 4.5771 | 14 | 0 | 4.3204 | 16 | 0 | 4.2377 |
| | | 0.10 | 7 | 1 | 3.4162 | 11 | 0 | 3.1941 | 13 | 0 | 3.1412 |
| 0.5 | 1 | 0.05 | 16 | 1 | 10.4744 | 18 | 0 | 9.8396 | 18 | 0 | 9.5066 |
| | | 0.075 | 12 | 1 | 6.9922 | 16 | 0 | 6.7514 | 18 | 0 | 6.6736 |
| | | 0.10 | 9 | 1 | 5.2021 | 13 | 0 | 5.0391 | 15 | 0 | 4.9872 |
| | 2 | 0.05 | 16 | 1 | 10.4744 | 18 | 0 | 9.4846 | 18 | 0 | 8.9179 |
| | | 0.075 | 12 | 1 | 6.9922 | 18 | 0 | 6.6718 | 18 | 0 | 6.3756 |
| | | 0.10 | 9 | 1 | 5.2021 | 15 | 0 | 4.9859 | 17 | 0 | 4.8770 |
| $r = 0.1$ | | | | | | | | | | | |
| 1 | 1 | 0.05 | 11 | 1 | 6.7993 | 13 | -1 | 6.2868 | 15 | 0 | 6.2558 |
| | | 0.075 | 8 | 1 | 4.5144 | 9 | -1 | 4.0746 | 11 | 0 | 4.1050 |
| | | 0.10 | 6 | 1 | 3.3179 | 8 | -1 | 3.0924 | 9 | 0 | 3.0543 |
| | 2 | 0.05 | 11 | 1 | 6.7993 | 16 | 0 | 6.3466 | 16 | 0 | 6.0524 |
| | | 0.075 | 8 | 1 | 4.5144 | 11 | 0 | 4.0958 | 12 | 0 | 3.9632 |
| | | 0.10 | 6 | 1 | 3.3179 | 9 | 0 | 3.0464 | 9 | 0 | 2.8522 |
| 0.5 | 1 | 0.05 | 15 | 1 | 10.3849 | 18 | 0 | 9.8396 | 18 | 0 | 9.5066 |
| | | 0.075 | 11 | 1 | 6.9260 | 14 | 0 | 6.5952 | 15 | 0 | 6.4593 |
| | | 0.10 | 9 | 1 | 5.2021 | 11 | 0 | 4.8806 | 12 | 0 | 4.7810 |
| | 2 | 0.05 | 15 | 1 | 10.3849 | 18 | 0 | 9.4846 | 18 | 0 | 8.9179 |
| | | 0.075 | 11 | 1 | 6.9260 | 15 | 0 | 6.4505 | 17 | 0 | 6.2854 |
| | | 0.10 | 9 | 1 | 5.2021 | 12 | 0 | 4.7729 | 14 | 0 | 4.6623 |
| $r = 0.25$ | | | | | | | | | | | |
| 1 | 1 | 0.05 | 10 | 1 | 6.6494 | 10 | -1 | 5.7733 | 10 | 0 | 5.3707 |
| | | 0.075 | 7 | 1 | 4.3540 | 7 | -1 | 3.6871 | 7 | 0 | 3.4490 |
| | | 0.10 | 5 | 1 | 3.0849 | 5 | -1 | 2.5601 | 5 | 0 | 2.4048 |
| | 2 | 0.05 | 10 | 1 | 6.6494 | 10 | -1 | 5.4871 | 10 | 0 | 5.0456 |
| | | 0.075 | 7 | 1 | 4.3540 | 7 | -1 | 3.4972 | 7 | 0 | 3.2007 |
| | | 0.10 | 5 | 1 | 3.0849 | 5 | -1 | 2.4237 | 5 | 0 | 2.2077 |
| 0.5 | 1 | 0.05 | 14 | 1 | 10.2076 | 14 | -1 | 9.0266 | 15 | 0 | 8.8760 |
| | | 0.075 | 10 | 1 | 6.7681 | 10 | -1 | 5.9058 | 11 | 0 | 5.8309 |
| | | 0.10 | 8 | 1 | 5.0649 | 8 | -1 | 4.3761 | 8 | 0 | 4.1177 |
| | 2 | 0.05 | 14 | 1 | 10.2076 | 16 | 0 | 9.0887 | 16 | 0 | 8.4950 |
| | | 0.075 | 10 | 1 | 6.7681 | 11 | 0 | 5.8022 | 11 | 0 | 5.3443 |
| | | 0.10 | 8 | 1 | 5.0649 | 8 | 0 | 4.0903 | 8 | 0 | 3.7190 |

$\mathcal{I} = 0$ to potentially no values of $C$, so that exhaustive service becomes the best choice. Not surprisingly, exhaustive service performs the best over these ranges when $p_{>0} = 1$.

# Chapter 3

# A 2-Class Maintenance Model with Dynamic Server Behaviour

## 3.1 Discussion of Literature

In Chapter 2, we investigated a closed queueing network tracking a finite population of machines which alternated between being functional or broken. Broken machines were assigned into one of two classes, and a single mechanic tended to the two queues as if in a polling system under either an exhaustive, non-preemptive priority, or preemptive resume priority service policy. Among the literature discussed in Section 2.1 were the works of Gross et al. [42] and Madu [67], who considered closed queueing networks of machines that can suffer two levels of failures having different service requirements. Both of these models allowed for an inventory of spares, which is often referred to as a maintenance float. In a system with a float, any excess functional machines when the system is at full capacity are turned off, and not at risk of failure, but are able to replace working machines that suffer failures. Spare machines incur their own costs of acquisition and upkeep, however they can be used to improve the average performance of the system as a whole. Taking inspiration from these works, within this chapter we extend the previous model to allow the existence of a maintenance float. We demonstrate within Section 3.5.1 a situation where it is optimal to use a float when spare machines come at a lower cost than increasing the maximum capacity of working machines.

Of course, there are other examples in the literature concerning maintenance float and inventory problems. For example, Lin et al. [63] studied a closed queueing maintenance network of $N$ machines across $M$ stations where each station was either a work or repair station, and each machine belonged to a specific work station (where it had to return, following its repair). Their interest was in finding the expected number of machines at each station, so as to select an optimal maintenance float. Liang et al. [62] investigated a system of $r$ fleets of machines, with the $i^{\text{th}}$ fleet having a capacity of $N_i$ working machines and a float of $S_i$ machines, and they assumed exponential failure rates for their machines depending on their respective fleets. Two versions of their model were considered, where every fleet had their own single mechanic, or where there existed a centralized repair shop with a single mechanic (who had a higher rate of repair) that was responsible for all fleets. In the latter case, they compared FCFS service, preemptive resume priority, and their own Myopic($\mathbf{R}$) policy (which looked at a future time point and compared differences in cost given different present decisions on what machine to serve) against the optimal policy determined through the use of a Markov decision process, and found their Myopic($\mathbf{R}$) policy comparable to the optimal policy.

Buyukkramikli et al. [25] also considered a maintenance system using an inventory of spare machines. In their model, the maintenance service provider was responsible for their operating costs as well as the costs of lost business for their client when the system was down. By holding an inventory of spare parts, they were able to immediately replace broken parts while they were undergoing repairs. Two versions of the model were considered, one with a permanent service capacity (i.e., service rate) and one with a two-level service capacity. At periodic intervals in the two-level case, the decision was made to invest (i.e., pay a higher cost per unit time) in the higher service capacity if the number of broken parts waiting to be repaired was beyond a selected threshold. The options to not require as high of an inventory of spare parts (allowing lower holding costs), and to have a lower base service capacity available in the two-level service capacity model, proved very effective in their numerical studies at lowering the optimal (minimum) costs incurred by the maintenance service provider. Kim and Dshalalow [53] studied the maintenance of machines within a production system with inventories of reserve and super reserve machines. The reserve machines were used to take the place of the working machines as they failed, until the time when all standard and reserve machines were working. At this point, the reserve machines were blocked and the super reserve machines took over, while the server went on vacation until a sufficient number of failures were observed to reduce the number of working machines without having a super reserve machine available to take its place. A cost function was put forth to be used to select optimal values for the maximum number of simultaneously working machines, the number of reserve machines, and the number of super reserve machines.

In addition to the inclusion of a maintenance float, this chapter expands that of Chapter 2 by generalizing the server's allowed behaviours. In particular, we allow for the probability of the server switching to the opposite queue at decision epochs after repair completions or machine failure instants to depend on both queue lengths, similar to Iravani et al. [48] and Liang et al. [62] within the context of using Markov decision processes to find the optimal server behaviour. This dynamic behaviour contains as special cases the exhaustive, preemptive resume priority, and non-preemptive priority service policies, as well as the $(a, b)$ threshold and smart Bernoulli policies which we introduce in detail in Section 3.3.2.

A threshold policy may be used in place of a preemptive or non-preemptive priority policy as a more tunable way to optimize a polling model by assigning priority to one queue over another. For instance, a threshold may be used by a server to meet a required level of service to higher priority real-time customers while minimizing the hindrance to their ability to serve non-real-time data. Lee and Sungupta [60] analyzed a 2-queue $M/G/1$ polling model where the server follows a 1-limited service discipline at both queues unless the queue length of the class-1 queue exceeds its threshold (at which point it is granted non-preemptive priority). Their model was work conserving in that a queue is served exhaustively if the other is empty. Boxma et al. [21] and Avram and Gómez-Corral [9] both researched 2-queue $M/M/1$ polling models, with the former allowing preemptive or non-preemptive priority to class 1 if its length reached its threshold, and the latter allowing only preemptive priority in the same situation.

A different take on threshold models can allow for a threshold to be set on both queues, so that the server knows to change their position if the opposite queue length gets too long while their current queue is below its corresponding threshold. Avrachenkov et al. [8] and Perel and Yechiali [74] both consider versions of a 2-queue $M/M/1$ polling model with this type of threshold policy for switching. In both works, work conserving and non-work conserving variants are proposed which determine the behaviour of the server after emptying their current queue. In the work conserving version, the server will switch to serve the opposite queue even if its length does not meet its threshold, while in the non-work conserving version they will idle until the opposite queue reaches its threshold or until another arrival is observed at their current queue.

Of course, a Bernoulli service policy (first introduced in the context of a $GI/G/1$ vacation model by Keilson and Servi [50]), which generalizes the exhaustive and 1-limited service policies, can also be used to optimize a polling model (e.g., Blanc and van der Mei [14]). Specifically, a server following a Bernoulli policy serves at least one customer per visit to a queue and assigns varying importance to each queue by way of a class-dependent probability (which may be varied) of the server initiating another service after a completion (should their queue be non-empty) rather than switching away. In Section 3.5.3, we argue for the optimality of setting one of our smart Bernoulli probabilities to 1 as in Blanc and van der Mei [14], reducing to a 2-queue polling model with exhaustive service at one queue and smart Bernoulli at the other. For an example of a 2-class polling model with exhaustive and the standard Bernoulli policy, one can refer to Weststrate and van der Mei [95]. For examples of Bernoulli service in a polling model with a general number of queues, with or without switchover times, see Blanc [11, 12]. Some other examples of papers which consider Bernoulli service disciplines are Boxma [18], Ramaswamy and Servi [78], Resing [79], and Servi [85]. We close this subsection by remarking that a majority of the work within this chapter may be found in Granville and Drekic [41].

## 3.2 Model Assumptions

We consider a maintenance system of $C + f$ identical machines, where $C \in \mathbb{Z}^+$ is the system's capacity, or the cap on how many machines may be in use at once (and hence at risk of failure), and $f \in \mathbb{N}$ denotes the number of machines in the maintenance float. The float provides an extra inventory of functional machines that replace machines that are taken down for repair after suffering a failure. It is assumed that a machine is not at risk of failure while turned off and stored in the float, and that they can instantaneously be put to use and turned on when needed. Following a machine repair, it is instantly turned on if the number of working machines immediately prior to the repair completion was less than $C$; otherwise, it is stored in the maintenance float.

The system is modelled as a 2-class polling model attended to by a lone mechanic (or server), where each class represents a grouping of one or more types of failure, and the service time distributions for each type of failure are allowed to be different. Let $\alpha_i$, $i = 1, 2$, be the total exponential class-$i$ failure rate, such that each machine, when turned on, has an effective failure rate of $\alpha = \alpha_1 + \alpha_2$. It is assumed that the failure times of machines are independent, machines fail individually, and a machine may only suffer one type of failure at a time. This last assumption may be worked around if the types of failure are within the same class by defining a combination of failures as a new type of failure (to be included in the same class).

Upon experiencing a class-$i$ failure (and being labelled as a *class-i machine* until it is repaired), a class-$i$ machine waits in the $i^{\text{th}}$ queue to receive service on a FCFS basis with respect to other class-$i$ machines in the same queue. To contrast the two classes of failures, we denote functional machines (either in use or stored in the float) as being of class 0. When every machine is class 0, rather than waiting at class 1 or class 2, the mechanic moves to a neutral third location, similarly named class 0.

It is assumed that class-$i$ service times are strictly positive and follow a continuous phase-type distribution with representation $Ser_i \sim \mathrm{PH}_{b_i}(\underline{\beta}_i, B_i)$. This is inherently a more restrictive assumption than generally distributed service times, although it is possible to approximate a (non-negative) non-phase-type distribution by fitting a phase-type one (most notably, via the classic EM Algorithm outlined by Asmussen et al. [7]). However, phase-types have a difficult time approximating some distributions well (particularly heavy-tailed ones), and increasing the number of phases to improve the fit can introduce computational issues due to the impact on the size of the state space of the model. We will apply the algorithm of Asmussen et al. [7] to approximate a heavy-tailed log-normal distribution to be used within the example in Section 3.5.3.

Fortunately, phase-type distributions do have many appealing features. Since phase-type distributions are closed under finite mixtures, it is straightforward to construct the underlying class-$i$ service time distribution from the individual continuous phase-type distributions corresponding to each type of failure within the same class. Depending on the assigned behaviour of the mechanic, it may be possible for a service time to be interrupted. In these cases, the service progress is not lost as the service phase is tracked to allow the mechanic to resume service where it left off, after eventually returning to that queue. Each service time is assumed to be independent of other services, as well as machine failure times.

Similarly, the time it takes the mechanic to 'switch' from class $j$ to class $i$ (referred to as a *class-$i$ switch-in*) is assumed to follow a continuous phase-type distribution with representation $\mathrm{PH}_{s_i}(\underline{\gamma}_{ji}, S_i)$, where the rate matrix $S_i$ depends only on the destination class, while the initial probability row vector $\underline{\gamma}_{ji}$ may also depend on the departure class. Switch-ins are also assumed to be independent of other switch-ins, as well as machine service and failure times. A switch-in having positive duration may, for example, represent any combination of the times necessary for the mechanic to change their instruments, retrieve spare parts, or physically relocate themselves to a different queue. If the time required to complete these tasks not directly related to serving an individual machine are insignificant, then it may make sense to allow the switch-in times to be identically zero. We let $\gamma_{ji}^{[0]} = 1 - \underline{\gamma}_{ji}\underline{e}'$ denote the probability of a class-$i$ switch-in (from class $j$) being equal to zero in duration.

As the mechanic may be allowed to preempt a switch-in within this system (if, say, one class has higher priority over the other at a given combination of queue lengths), we make the assumption that switching out of a class-$i$ switch-in is the same as beginning a switch-in after the completion (or preemption) of a class-$i$ service time. That is, for example, if class 1 has a higher priority than class 2 and the mechanic observes a class-1 failure while conducting a switch from class 0 to class 2, then they will start a new class-1 switch-in with initial probability vector $\underline{\gamma}_{21}$. We remark that a class-0 switch-in will always be interrupted if the mechanic observes a machine failure from either class.

We provide a depiction of the maintenance system as described above in Figure 3.1. Note that the notation $X_1$, $X_2$, and $L$ are as they will be defined in Section 3.3.1, representing the first and second queue lengths, and the position of the server, respectively. Machines are represented by solid black circles, while slots that machines may take within class 0 (whether to be put in use or in the maintenance float) are represented by empty circles. Similarly, the larger solid grey circle and dashed empty circles represent current and potential locations where the server either works or idles, with the grey circle in this example implying that the mechanic is currently serving class-2 machines. As defined above, the distribution of the time between service completions is $\mathrm{PH}_{b_i}(\underline{\beta}_i, B_i)$ (in this example, we would have $i = 2$),

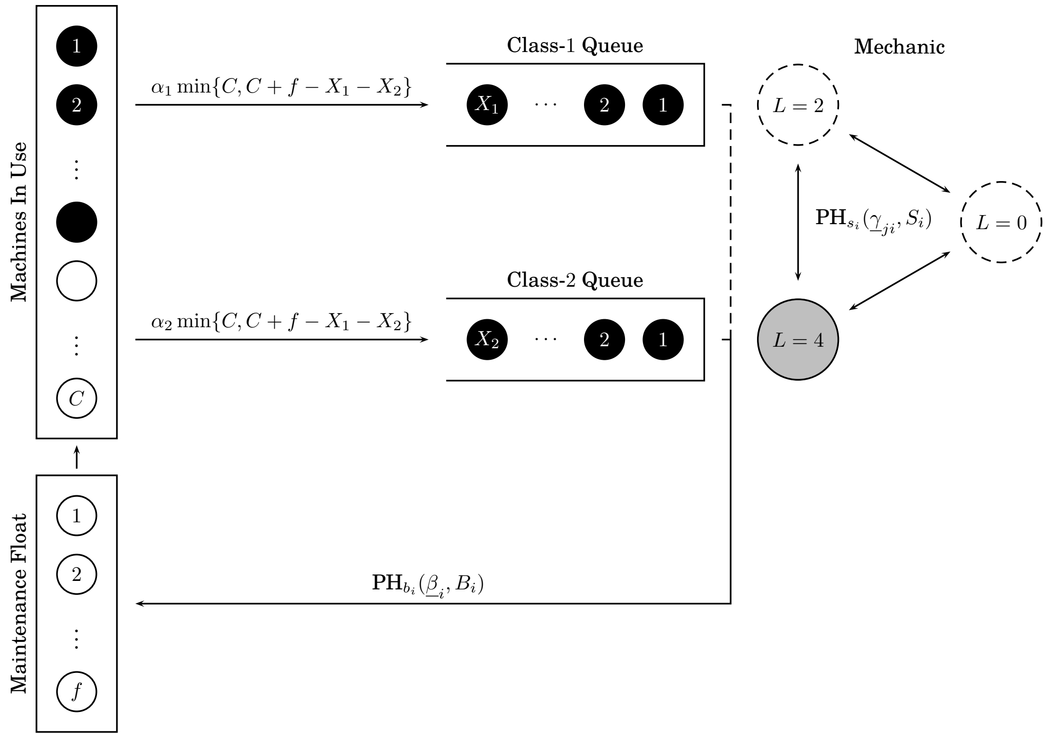
Machines In Use
1
2
$\vdots$
$\vdots$
$C$
Maintenance Float
1
2
$\vdots$
$f$
$\alpha_1 \min\{C, C + f - X_1 - X_2\}$
$\alpha_2 \min\{C, C + f - X_1 - X_2\}$
Class-1 Queue
$X_1$
$\cdots$
2
1
Class-2 Queue
$X_2$
$\cdots$
2
1
Mechanic
$L = 2$
$L = 0$
$L = 4$
$\text{PH}_{s_i}(\underline{\gamma}_{ji}, S_i)$
$\text{PH}_{b_i}(\underline{\beta}_i, B_i)$

Figure 3.1: Depiction of the maintenance system with a maintenance float and the server at queue 2.

and if the server switches between the three locations, the time to complete the switch has a $\text{PH}_{s_i}(\underline{\gamma}_{ji}, S_i)$ distribution. Repaired machines are brought to the maintenance float, where they will automatically be put to use if there are any open slots for functional machines. Figure 3.1 does assume that a float exists (i.e., $f \geq 1$), but we do in fact allow the choice of $f = 0$. In the $f = 0$ case, the diagram would change by way of having no float, and repaired machines would automatically be put to use.

A defining feature of polling models is the chosen service policy which dictates the server's behaviour. In this model, we allow our mechanic be dynamic, whose decision to start a switch-in (i.e., the probability of deciding to switch) may depend on both queue lengths as well as what type of event is causing the server to make a decision, namely after a service completion (when the other queue has a positive length), or after observing an arrival to the opposite queue during a switch-in or a service. As these decision probabilities are state-dependent, we must first define the state space of the Markov chain describing this system before constructing the decision probability matrices.

## 3.3 Model Construction and Analysis

### 3.3.1 State Space and State-Dependent Decision Probabilities

In order to model this maintenance system without restricting the server's behaviour, we must track six variables within our state space, using the CTMC

$$\{(X_1(t), X_2(t), L(t), Y(t), Y_1(t), Y_2(t)), t \geq 0\}.$$

As customers from either class may, in general, experience service interruptions, this is similar to a combination of the preemptive priority models considered in Sections 2.4.1 and 2.4.2. Here, $X_1(t) \in \{0, 1, \ldots, C + f\}$ is the length of the class-1 queue and is treated as the level of the process. Next, $X_2(t) \in \{0, 1, \ldots, C + f - X_1(t)\}$ is the length of the class-2 queue. $L(t) \in \{0, 1, 2, 3, 4, 5\}$ denotes the location of the server (0: idle at class 0; 1: switching into class 1; 2: serving class 1; 3: switching into class 2; 4: serving class 2; 5: switching into class 0). $Y(\mathrm{t})$ denotes the phase of a switch-in time or takes the value of 0 when the mechanic is either idle or repairing a machine, i.e.,

$$Y(t) \in \Omega_Y(L(t)) = \begin{cases} \{0\} & \text{, if } L(t) = 0, \\ \{1, 2, \ldots, s_1\} & \text{, if } L(t) = 1, \\ \{0\} & \text{, if } L(t) = 2, \\ \{1, 2, \ldots, s_2\} & \text{, if } L(t) = 3, \\ \{0\} & \text{, if } L(t) = 4, \\ \{1, 2, \ldots, s_0\} & \text{, if } L(t) = 5. \end{cases}$$

Lastly, $Y_1(t)$ and $Y_2(t)$ are the current phases of service of the class-1 and class-2 machines leading their respective queues. $Y_i(t)$ takes on a value of zero if the $i^{\text{th}}$ queue is empty, so that

$$Y_i(t) \in \Omega_{Y_i}(X_i(t)) = \begin{cases} \{0\} & \text{, if } X_i(t) = 0, \\ \{1, 2, \ldots, b_i\} & \text{, if } X_i(t) \geq 1. \end{cases}$$

Note that this variable is initialized as soon as $X_i(t)$ changes from 0 to 1 (after observing a class-$i$ failure), $i = 1, 2$, which is in general not the same time as when the customer's service actually begins.

With the above notation in place, we can now define the decision probability matrices. As mentioned previously, we categorize decision epochs into one of three types, with the first type occurring after a service completion. Define $\mathcal{P}^{1S}_{m,n}$ as the probability of initiating a class-1 switch-in (from class 2) immediately after a class-2 service completion that reduces $X_2(t)$ from $n+1$ to $n$, when $X_1(t) = m$. For ease of presentation (and storage), we let

$$\mathcal{P}^{1S} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ \vdots \\ C{+}f{-}3 \\ C{+}f{-}2 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & \cdots & C{+}f{-}3 & C{+}f{-}2 \end{array} \\ \left[ \begin{array}{cccccc} \mathcal{P}^{1S}_{1,1} & \mathcal{P}^{1S}_{1,2} & \mathcal{P}^{1S}_{1,3} & \cdots & \mathcal{P}^{1S}_{1,C+f-3} & \mathcal{P}^{1S}_{1,C+f-2} \\ \mathcal{P}^{1S}_{2,1} & \mathcal{P}^{1S}_{2,2} & \mathcal{P}^{1S}_{2,3} & \cdots & \mathcal{P}^{1S}_{2,C+f-3} & 0 \\ \mathcal{P}^{1S}_{3,1} & \mathcal{P}^{1S}_{3,2} & \mathcal{P}^{1S}_{3,3} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ \mathcal{P}^{1S}_{C+f-3,1} & \mathcal{P}^{1S}_{C+f-3,2} & 0 & \cdots & 0 & 0 \\ \mathcal{P}^{1S}_{C+f-2,1} & 0 & 0 & \cdots & 0 & 0 \end{array} \right] \end{array}.$$

Note that we do not need to define probabilities where $X_1(t) + X_2(t) = m + n = C + f$, since there must be at least one functional machine after a service completion, and we do not consider probabilities for $m = 0$ or $n = 0$, as we make the assumption that the mechanic will always choose to serve the class having a non-zero queue length should the other queue be empty. A corresponding matrix $\mathcal{P}^{2S}$ is also constructed in the same way, such that $\mathcal{P}^{2S}_{m,n}$ is the probability of switching to serve class 2 after a class-1 service completion which reduces $X_1(t)$ from $m+1$ to $m$, when $X_2(t) = n$.

Next, we define $\mathcal{P}^{1P}_{m,n}$ ($\mathcal{P}^{2P}_{m,n}$) and $\mathcal{P}^{1N}_{m,n}$ ($\mathcal{P}^{2N}_{m,n}$) to be the probabilities of the server initiating a class-1 (class-2) switch-in after observing a class-1 (class-2) failure that results in $(X_1(t), X_2(t)) = (m, n)$ after said failure when $L(t) = 4$ ($L(t) = 2$) or $L(t) = 3$ ($L(t) = 1$) immediately prior to the failure epoch, respectively. We distinguish these probabilities with a $P$ or $N$ to denote the fact that they represent switch-ins that are either *preemptive* or *non-preemptive* in nature, with respect to service times of the opposite class. We now let

$$\mathcal{P}^{1P} = \begin{array}{c|ccccccc|}
 & 1 & 2 & 3 & \cdots & C+f-3 & C+f-2 & C+f-1 \\
1 & \mathcal{P}^{1P}_{1,1} & \mathcal{P}^{1P}_{1,2} & \mathcal{P}^{1P}_{1,3} & \cdots & \mathcal{P}^{1P}_{1,C+f-3} & \mathcal{P}^{1P}_{1,C+f-2} & \mathcal{P}^{1P}_{1,C+f-1} \\
2 & \mathcal{P}^{1P}_{2,1} & \mathcal{P}^{1P}_{2,2} & \mathcal{P}^{1P}_{2,3} & \cdots & \mathcal{P}^{1P}_{2,C+f-3} & \mathcal{P}^{1P}_{2,C+f-2} & 0 \\
3 & \mathcal{P}^{1P}_{3,1} & \mathcal{P}^{1P}_{3,2} & \mathcal{P}^{1P}_{3,3} & \cdots & \mathcal{P}^{1P}_{3,C+f-3} & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots \\
C+f-3 & \mathcal{P}^{1P}_{C+f-3,1} & \mathcal{P}^{1P}_{C+f-3,2} & \mathcal{P}^{1P}_{C+f-3,3} & \cdots & 0 & 0 & 0 \\
C+f-2 & \mathcal{P}^{1P}_{C+f-2,1} & \mathcal{P}^{1P}_{C+f-2,2} & 0 & \cdots & 0 & 0 & 0 \\
C+f-1 & \mathcal{P}^{1P}_{C+f-1,1} & 0 & 0 & \cdots & 0 & 0 & 0
\end{array},$$

and similarly define $\mathcal{P}^{1N}$, $\mathcal{P}^{2P}$, and $\mathcal{P}^{2N}$, containing the same ranges of indexed probabilities. In contrast to $\mathcal{P}^{1S}$ and $\mathcal{P}^{2S}$, we now must consider cases with $m+n = C+f$, since it is possible for there to be no functional machines after a failure. We still do not need to consider cases with either $m = 0$ or $n = 0$, since the event of observing an arrival to one queue (in the form of a machine failure) while switching into or serving at the other implies that both queue lengths are positive after the failure.

Finally, we define the class-1 adjusted decision probabilities

$$d_i^{[D,g]}(m,n) = \mathcal{P}^{iS}_{m+(C+f)-(D+g),n}, \; i = 1,2, \tag{3.1}$$

$$a_{i,p}^{[D,g]}(m,n) = \mathcal{P}^{iP}_{m+(C+f)-(D+g),n}, \; i = 1,2, \tag{3.2}$$

and

$$a_i^{[D,g]}(m,n) = \mathcal{P}^{iN}_{m+(C+f)-(D+g),n}, \; i = 1,2, \tag{3.3}$$

such that, for example, $d_2^{[C-l,0]}(m,n) = \mathcal{P}^{2S}_{m+l+f,n}$ and $a_1^{[C,f]}(m,n) = \mathcal{P}^{1N}_{m,n}$. Note that the inclusion of '$(C+f)$' in the subscripts above is treated as a constant (i.e., independent of the superscript of generator blocks to be defined in Section 3.3.3), allowing us to accurately determine the length of the class-1 queue as we reduce the effective number of machines from $[C,f]$ to $[D,g]$ in the system as part of the sojourn time analysis in Section 3.3.4.

### 3.3.2 Select Service Policies and Their Decision Probability Matrices

Within the numerical examples in Sections 3.4 and 3.5, we examine several service policies of interest which we are able to construct from specific combinations of decision probabilities.

Before specifying these cases, we define $\mathcal{A}$ as the matrix $\mathcal{P}^{1P}$ if we let $\mathcal{P}^{1P}_{m,n} = 1$, $m = 1, 2, \ldots, C + f - 1$, $n = 1, 2, \ldots, C + f - m$. That is, $\mathcal{A}$ has the same dimension and structure as the four failure instant decision probability matrices, but with each probability set equal to 1. Similarly, define $\mathcal{D}$ as the matrix $\mathcal{P}^{1S}$ with $\mathcal{P}^{1S}_{m,n} = 1$, $m = 1, 2, \ldots, C + f - 2$, $n = 1, 2, \ldots, C + f - m - 1$. Finally, for $j \in \mathbb{Z}^+$ and $i = 1, 2, \ldots, j + 1$, let

$$\mathcal{T}_i^{[j]} = \begin{array}{c|cccccccc|}
 & 1 & 2 & \cdots & j-i & j+1-i & j+2-i & \cdots & j-1 & j \\
\hline
1 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\
2 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots & \vdots \\
i-1 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\
i & 1 & 1 & \cdots & 1 & 1 & 0 & \cdots & 0 & 0 \\
i+1 & 1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots & \vdots \\
j-1 & 1 & 1 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\
j & 1 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\
\hline
\end{array},$$

such that in its boundary cases,

$$\mathcal{T}_i^{[j]} = \begin{cases} \mathcal{A} & \text{, if } i = 1, j = C + f - 1, \\ \mathcal{D} & \text{, if } i = 1, j = C + f - 2, \\ \mathbf{0} & \text{, if } i > j, \end{cases} \tag{3.4}$$

where $\mathbf{0}$ denotes an appropriately dimensioned matrix of zeroes, which in this case has dimension $j \times j$.

We now discuss our service policies of interest, whose switch-in decision probability matrices are specified in Table 3.1. The first service policy we consider is the classic *exhaustive* service policy, where the server remains at a particular queue until it empties, at which time a switch to the other queue is made, or for our model specifically, to class 0 if $X_1(t) = X_2(t) = 0$ at this time. Since the server will never leave a queue while it has a positive length, all decision probabilities must be zero.

Next, we have a pair of priority policies wherein the mechanic prefers to serve one class of failures before the other. We present the class-1 priority policies here, while the class-2 priority policies may be obtained by simply interchanging the class-1 and class-2 decision probabilities. For class-1 *non-preemptive priority*, the server will always immediately begin a class-1 switch-in upon observing a class-1 failure to an empty queue (note that the server is only allowed to leave queue 1 once $X_1(t) = 0$) so long as they do not have to interrupt, or preempt, a class-2 service time. Thus, the decision probabilities when conducting a class-2 switch-in, or after completing a class-2 service, are all one, whereas the decision probabilities during a class-2 service time are zero. In contrast, the *preemptive resume priority* policy gives the server permission to interrupt a service time, which will later be resumed with no work lost, so the decision probabilities during a class-2 service time are also set equal to one. We remark that we chose to let $\mathcal{P}^{1S} = \mathcal{D}$ in the preemptive case, even though it is not possible to observe a class-2 service completion when $X_1(t) > 0$ (and hence, in practice, these probabilities will never be checked by the CTMC).

A threshold policy is a modification of standard priority policies, in that a class's higher priority is conditional on it having a queue length equaling or exceeding a particular class-dependent threshold. As we are already considering priority policies that are both preemptive

Table 3.1: Switch-in decision probability matrices for select service policies.

| Service Policy | $\mathcal{P}^{1S}$ | $\mathcal{P}^{1P}$ | $\mathcal{P}^{1N}$ | $\mathcal{P}^{2S}$ | $\mathcal{P}^{2P}$ | $\mathcal{P}^{2N}$ |
|---|---|---|---|---|---|---|
| Exhaustive | $\mathbf{0}$ | $\mathbf{0}$ | $\mathbf{0}$ | $\mathbf{0}$ | $\mathbf{0}$ | $\mathbf{0}$ |
| Class-1 Non-preemptive Priority | $\mathcal{D}$ | $\mathbf{0}$ | $\mathcal{A}$ | $\mathbf{0}$ | $\mathbf{0}$ | $\mathbf{0}$ |
| Class-1 Preemptive Resume Priority | $\mathcal{D}$ | $\mathcal{A}$ | $\mathcal{A}$ | $\mathbf{0}$ | $\mathbf{0}$ | $\mathbf{0}$ |
| Class-1 $(a, b)$ Threshold | $\mathcal{T}_a^{[C+f-2]}$ | $\mathcal{T}_b^{[C+f-1]}$ | $\mathcal{T}_a^{[C+f-1]}$ | $\mathbf{0}$ | $\mathbf{0}$ | $\mathbf{0}$ |
| $(p_1^{\text{SB}}, p_2^{\text{SB}})$ Smart Bernoulli | $(1 - p_2^{\text{SB}})\mathcal{D}$ | $\mathbf{0}$ | $\mathbf{0}$ | $(1 - p_1^{\text{SB}})\mathcal{D}$ | $\mathbf{0}$ | $\mathbf{0}$ |

and non-preemptive in nature, we elect to use a variant of the threshold policy which can assign non-preemptive priority to a class after reaching a threshold ($a$), and then preemptive resume priority to a class after reaching another threshold ($b$) that is equal to or greater than the non-preemptive threshold. That is, if $X_1(t) < a$, then the server acts as if under an exhaustive policy, if $a \leq X_1(t) < b$, the server acts as if under a class-1 non-preemptive policy, and if $b \leq X_1(t)$, the server acts as if under a class-1 preemptive resume priority policy. We refer to this variant as an $(a, b)$ *threshold* policy.

In order to handle the activation of priority, we make use of the above $\mathcal{T}_i^{[j]}$ matrices which change from having decision probabilities of zero to probabilities of one once $X_1(t) \geq i$ (i.e., for row $i$ and below). If we instead wanted to use a class-2 threshold policy, we would use transposes of these matrices, $\left(\mathcal{T}_i^{[j]}\right)'$, so that the policy would adjust after observing $X_2(t) \geq i$ (i.e., for column $i$ and to the right). As implied by Equation (3.4) and Table 3.1, an $(a, b)$ threshold policy can recover the exhaustive or priority policies. In fact, it can also represent a non-preemptive threshold policy if we let $b = C + f$, or a preemptive resume threshold policy if we let $b = a$.

Lastly, we consider a modification of the Bernoulli service discipline introduced by Keilson and Servi [50]. In the original discipline, after every service completion, the server would either switch away or go on vacation (in the case of a single queue system) depending on the result of an independent Bernoulli trial having a fixed class-dependent probability. In our model, we are assuming that the mechanic has full information concerning queue lengths, and as such would not be inclined to switch away from a queue without emptying it if the opposite queue has no machines waiting to be serviced. Therefore, we implement a modified policy that we refer to as $(p_1^{\text{SB}}, p_2^{\text{SB}})$ *smart Bernoulli*, or simply smart Bernoulli, which only conducts a class-dependent Bernoulli trial having probability $p_i^{\text{SB}}$ of starting another class-$i$ service, $i = 1, 2$, rather than switching away to the opposite queue, if the opposite queue has a positive length. Hence, under this policy, the only decisions the server has to make are at service completions, and these decisions always have the same class-dependent probability for each combination of queue lengths. Finally, we remark that if we let $p_1^{\text{SB}} = p_2^{\text{SB}} = 1$, then the server never leaves a queue until it is empty and we recover the exhaustive service policy.

### 3.3.3 Steady-State Probabilities

We are able to solve for the steady-state probabilities by representing the system as a level-dependent QBD process, taking the length of the class-1 queue, $X_1$, as the level of the process. First of all, let $\pi_{m,n,l,y,y_1,y_2}$ be the steady-state probability of observing the CTMC in state $(X_1(t), X_2(t), L(t), Y(t), Y_1(t), Y_2(t)) = (m, n, l, y, y_1, y_2)$, where the variables take on values from their supports defined in Section 3.3.1. Next, we order the steady-state probabilities into the row vector

$$\underline{\pi} = (\underline{\pi}_0, \underline{\pi}_1, \ldots, \underline{\pi}_{C+f}), \tag{3.5}$$

where

$$\underline{\pi}_m = (\underline{\pi}_{m,0}, \underline{\pi}_{m,1}, \cdots, \underline{\pi}_{m,C+f-m})$$

contains the ordered steady-state probabilities for level $m$, $m = 0, 1, \ldots, C + f$. For level 0,

$$\underline{\pi}_{0,0} = (\pi_{0,0,0,0,0,0}, \pi_{0,0,5,1,0,0}, \cdots, \pi_{0,0,5,s_0,0,0})$$

has length $1 + s_0$, and

$$\begin{aligned}\underline{\pi}_{0,n} = (&\pi_{0,n,3,1,0,1}, \cdots, \pi_{0,n,3,1,0,b_2}, \pi_{0,n,3,2,0,1}, \cdots, \pi_{0,n,3,s_2,0,b_2}, \\ &\pi_{0,n,4,0,0,1}, \cdots, \pi_{0,n,4,0,0,b_2})\end{aligned}$$

has length $(s_2+1)b_2$ for $n = 1, 2, \ldots, C+f$, resulting in $1 + s_0 + (C+f)(s_2+1)b_2$ total states. For level $m = 1, 2, \ldots, C+f$,

$$\begin{aligned}\underline{\pi}_{m,0} = (&\pi_{m,0,1,1,1,0}, \cdots, \pi_{m,0,1,1,b_1,0}, \pi_{m,0,1,2,1,0}, \cdots, \pi_{m,0,1,s_1,b_1,0}, \\ &\pi_{m,0,2,0,1,0}, \cdots, \pi_{m,0,2,0,b_1,0})\end{aligned}$$

has length $(s_1+1)b_1$, and for $m = 1, 2, \ldots, C+f-1$ and $n = 1, 2, \ldots, C+f-m$,

$$\begin{aligned}\underline{\pi}_{m,n} = (&\pi_{m,n,1,1,1,1}, \cdots, \pi_{m,n,1,1,1,b_2}, \pi_{m,n,1,1,2,1}, \cdots, \pi_{m,n,1,1,b_1,b_2}, \pi_{m,n,1,2,1,1}, \cdots, \\ &\pi_{m,n,1,s_1,b_1,b_2}, \pi_{m,n,2,0,1,1}, \cdots, \pi_{m,n,2,0,1,b_2}, \pi_{m,n,2,0,2,1}, \cdots, \pi_{m,n,2,0,b_1,b_2}, \\ &\pi_{m,n,3,1,1,1}, \cdots, \pi_{m,n,3,1,1,b_2}, \pi_{m,n,3,1,2,1}, \cdots, \pi_{m,0,3,1,b_1,b_2}, \pi_{m,n,3,2,1,1}, \cdots, \\ &\pi_{m,n,3,s_2,b_1,b_2}, \pi_{m,n,4,0,1,1}, \cdots, \pi_{m,n,4,0,1,b_2}, \pi_{m,n,4,0,2,1}, \cdots, \pi_{m,n,4,0,b_1,b_2})\end{aligned}$$

has length $(s_1+s_2+2)b_1b_2$, resulting in $(s_1+1)b_1 + (C+f-m)(s_1+s_2+2)b_1b_2$ total states.

The corresponding infinitesimal generator $Q^{[C,f]}$ for this QBD process takes on the form

$$Q^{[C,f]} = \begin{array}{c|ccccccc|} & 0 & 1 & 2 & \cdots & C+f-2 & C+f-1 & C+f \\ 0 & Q^{[C,f]}_{0,0} & Q^{[C,f]}_{0,1} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 1 & Q^{[C,f]}_{1,0} & Q^{[C,f]}_{1,1} & Q^{[C,f]}_{1,2} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 2 & \mathbf{0} & Q^{[C,f]}_{2,1} & Q^{[C,f]}_{2,2} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ C+f-2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q^{[C,f]}_{C+f-2,C+f-2} & Q^{[C,f]}_{C+f-2,C+f-1} & \mathbf{0} \\ C+f-1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q^{[C,f]}_{C+f-1,C+f-2} & Q^{[C,f]}_{C+f-1,C+f-1} & Q^{[C,f]}_{C+f-1,C+f} \\ C+f & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & Q^{[C,f]}_{C+f,C+f-1} & Q^{[C,f]}_{C+f,C+f} \end{array}, \tag{3.6}$$

where each submatrix (or block) $Q^{[C,f]}_{i,j}$ contains all rates corresponding to state transitions where the level changes from $i$ to $j$. Here, we use superscript '$[C, f]$' to denote the sizes of the system's capacity ($C$) and maintenance float ($f$), in contrast to earlier where we allowed the second number in the superscript to denote the service discipline. We will make use of this version of our superscript notation in the sojourn time analysis of Section 3.3.4. As in our previous considered maintenance model, the steady-state probability row vector $\underline{\pi}$ can be solved by applying the level-dependent QBD algorithm from Section 1.2.6.

We conclude this subsection by specifying the constructed blocks of $Q^{[C,f]}$. To this end, we recall the following notation. We let $\otimes$ represent the standard Kronecker product operator, let $I_i$ be an $i \times i$ identity matrix, and let $\underline{e}_i$ be a row vector of ones having length $i$. In addition, we define $\underline{B}'_{0,i} = -B_i\underline{e}'$ and $\underline{S}'_{0,i} = -S_i\underline{e}'$ as the column vectors of absorption rates for the $\text{PH}_{b_i}(\underline{\beta}_i, B_i)$ distributed class-$i$ service times and $\text{PH}_{s_i}(\underline{\gamma}_{ji}, S_i)$ distributed class-$i$ switch-in times, respectively, and let $\underline{\gamma}_{ji}^{[+0]} = (\underline{\gamma}_{ji}, \gamma_{ji}^{[0]})$ be the concatenated probability vector joining the initial distribution of a class $j$ to class $i$ switch-in with the probability of the switch-in being zero in duration. Finally, let $\Delta_{m,n}^{[C,f]} = \min\{C, C+f-m-n\}$ denote the number of working machines when $X_1(t) = m$ and $X_2(t) = n$.

For levels $m = 0, 1, \ldots, C+f$, the main diagonal blocks of $Q^{[C,f]}$ are given by

$$Q_{m,m}^{[C,f]} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C+f-m-1 \\ C+f-m \end{array} \begin{array}{c} \begin{array}{cccccc} 0 & 1 & 2 & \cdots & C+f-m-1 & C+f-m \end{array} \\ \left[ \begin{array}{cccccc} Q_{m,m,0}^{[C,f]} & (UD)_{m,0}^{[C,f]} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ (LD)_{m,1}^{[C,f]} & Q_{m,m,1}^{[C,f]} & (UD)_{m,1}^{[C,f]} & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (LD)_{m,2}^{[C,f]} & Q_{m,m,2}^{[C,f]} & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q_{m,m,C+f-m-1}^{[C,f]} & (UD)_{m,C+f-m-1}^{[C,f]} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & (LD)_{m,C+f-m}^{[C,f]} & Q_{m,m,C+f-m}^{[C,f]} \end{array} \right] \end{array},$$

where for $m = 0$

$$Q_{0,0,0}^{[C,f]} = -C\alpha I_{1+s_0} + \left[ \begin{array}{cc} 0 & \underline{0}_{s_0} \\ \underline{S}'_{0,0} & S_0 \end{array} \right],$$

$$Q_{0,0,n}^{[C,f]} = -\Delta_{0,n}^{[C,f]}\alpha I_{(s_2+1)b_2} + \left[ \begin{array}{cc} S_2 \otimes I_{b_2} & \underline{S}'_{0,2} \otimes I_{b_2} \\ \mathbf{0} & B_2 \end{array} \right],\ n = 1, 2, \ldots, C+f,$$

$$(UD)_{0,0}^{[C,f]} = C\alpha_2\underline{e}'_{1+s_0}\underline{\gamma}_{02}^{[+0]} \otimes \underline{\beta}_2,$$

$$(UD)_{0,n}^{[C,f]} = \Delta_{0,n}^{[C,f]}\alpha_2 I_{(s_2+1)b_2},\ n = 1, 2, \ldots, C+f-1,$$

$$(LD)_{0,1}^{[C,f]} = \left[ \begin{array}{cc} \underline{0}'_{s_2b_2} & \mathbf{0} \\ \gamma_{20}^{[0]}\underline{B}'_{0,2} & \underline{B}'_{0,2}\underline{\gamma}_{20} \end{array} \right],$$

and

$$(LD)_{0,n}^{[C,f]} = \left[ \begin{array}{cc} \underline{0}'_{s_2b_2}\underline{0}_{s_2b_2} & \mathbf{0} \\ \mathbf{0} & \underline{B}'_{0,2}\underline{\beta}_2 \end{array} \right],\ n = 2, 3, \ldots, C+f,$$

while for $m = 1, 2, \ldots, C+f$,

$$Q_{m,m,0}^{[C,f]} = -\Delta_{m,0}^{[C,f]}\alpha I_{(s_1+1)b_1} + \left[ \begin{array}{cc} S_1 \otimes I_{b_1} & \underline{S}'_{0,1} \otimes I_{b_1} \\ \mathbf{0} & B_1 \end{array} \right],$$

and

$$Q_{m,m,n}^{[C,f]} = -\Delta_{m,n}^{[C,f]}\alpha I_{(s_1+s_2+2)b_1b_2} + \left[ \begin{array}{cccc} S_1 \otimes I_{b_1b_2} & \underline{S}'_{0,1} \otimes I_{b_1b_2} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & B_1 \otimes I_{b_2} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & S_2 \otimes I_{b_1b_2} & \underline{S}'_{0,2} \otimes I_{b_1b_2} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & I_{b_1} \otimes B_2 \end{array} \right]$$

for $n = 1, 2, \ldots, C+f-m$,

$$(UD)^{[C,f]}_{m,0} = \Delta^{[C,f]}_{m,0}\alpha_2 \left[ \begin{array}{ccc} (1-a^{[C,f]}_2(m,1))I_{s_1b_1} \otimes \underline{\beta}_2 & \mathbf{0} & a^{[C,f]}_2(m,1)\underline{e}'_{s_1}\underline{\gamma}^{[+0]}_{12} \otimes I_{b_1} \otimes \underline{\beta}_2 \\ \mathbf{0} & (1-a^{[C,f]}_{2,p}(m,1))I_{b_1} \otimes \underline{\beta}_2 & a^{[C,f]}_{2,p}(m,1)\underline{\gamma}^{[+0]}_{12} \otimes I_{b_1} \otimes \underline{\beta}_2 \end{array} \right],$$

and

$$(UD)^{[C,f]}_{m,n} = \Delta^{[C,f]}_{m,n}\alpha_2 \left[ \begin{array}{ccc} (1-a^{[C,f]}_2(m,n+1))I_{s_1b_1b_2} & \mathbf{0} & a^{[C,f]}_2(m,n+1)\underline{e}'_{s_1}\underline{\gamma}^{[+0]}_{12} \otimes I_{b_1b_2} \\ \mathbf{0} & (1-a^{[C,f]}_{2,p}(m,n+1))I_{b_1b_2} & a^{[C,f]}_{2,p}(m,n+1)\underline{\gamma}^{[+0]}_{12} \otimes I_{b_1b_2} \\ \mathbf{0} & \mathbf{0} & I_{(s_2+1)b_1b_2} \end{array} \right]$$

for $n = 1, 2, \ldots, C+f-m-1$, and

$$(LD)^{[C,f]}_{m,1} = \left[ \begin{array}{c} \underline{0}'_{(s_1+s_2+1)b_1b_2}\underline{0}_{(s_1+1)b_1} \\ \underline{\gamma}^{[+0]}_{21} \otimes I_{b_1} \otimes \underline{B}'_{0,2} \end{array} \right],$$

and

$$(LD)^{[C,f]}_{m,n} = \left[ \begin{array}{ccc} \underline{0}'_{(s_1+s_2+1)b_1b_2}\underline{0}_{(s_1+1)b_1b_2} & \underline{0}'_{(s_1+s_2+1)b_1b_2}\underline{0}_{s_2b_1b_2} & \underline{0}'_{(s_1+s_2+1)b_1b_2}\underline{0}_{b_1b_2} \\ d^{[C,f]}_1(m,n-1)\underline{\gamma}^{[+0]}_{21} \otimes I_{b_1} \otimes \underline{B}'_{0,2}\underline{\beta}_2 & \mathbf{0} & (1-d^{[C,f]}_1(m,n-1))I_{b_1} \otimes \underline{B}'_{0,2}\underline{\beta}_2 \end{array} \right]$$

for $n = 2, 3, \ldots, C+f-m$.

Next, for levels $m = 0, 1, \ldots, C+f-1$, the upper diagonal blocks of $Q^{[C,f]}$ have the form

$$Q^{[C,f]}_{m,m+1} = \begin{array}{c|ccccc|} & 0 & 1 & 2 & \cdots & C+f-m-1 \\ \hline 0 & Q^{[C,f]}_{m,m+1,0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ 1 & \mathbf{0} & Q^{[C,f]}_{m,m+1,1} & \mathbf{0} & \ddots & \mathbf{0} \\ 2 & \mathbf{0} & \mathbf{0} & Q^{[C,f]}_{m,m+1,2} & \ddots & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ C+f-m-1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q^{[C,f]}_{m,m+1,C+f-m-1} \\ C+f-m & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \end{array},$$

where for $m = 0$,

$$Q^{[C,f]}_{0,1,0} = C\alpha_1\underline{e}'_{1+s_0}\underline{\gamma}^{[+0]}_{01} \otimes \underline{\beta}_1,$$

and

$$Q^{[C,f]}_{0,1,n} = \Delta^{[C,f]}_{0,n}\alpha_1 \left[ \begin{array}{ccc} a^{[C,f]}_1(1,n)\underline{e}'_{s_2}\underline{\gamma}^{[+0]}_{21} \otimes \underline{\beta}_1 \otimes I_{b_2} & (1-a^{[C,f]}_1(1,n))I_{s_2} \otimes \underline{\beta}_1 \otimes I_{b_2} & \mathbf{0} \\ a^{[C,f]}_{1,p}(1,n)\underline{\gamma}^{[+0]}_{21} \otimes \underline{\beta}_1 \otimes I_{b_2} & \mathbf{0} & (1-a^{[C,f]}_{1,p}(1,n))\underline{\beta}_1 \otimes I_{b_2} \end{array} \right]$$

for $n = 1, 2, \ldots, C+f-1$, while for $m = 1, 2, \ldots, C+f-1$,

$$Q^{[C,f]}_{m,m+1,0} = \Delta^{[C,f]}_{m,0}\alpha_1 I_{(s_1+1)b_1},$$

and

$$Q^{[C,f]}_{m,m+1,n} = \Delta^{[C,f]}_{m,n}\alpha_1 \left[ \begin{array}{ccc} I_{(s_1+1)b_1b_2} & \mathbf{0} & \mathbf{0} \\ a^{[C,f]}_1(m+1,n)\underline{e}'_{s_2}\underline{\gamma}^{[+0]}_{21} \otimes I_{b_1b_2} & (1-a^{[C,f]}_1(m+1,n))I_{s_2b_1b_2} & \mathbf{0} \\ a^{[C,f]}_{1,p}(m+1,n)\underline{\gamma}^{[+0]}_{21} \otimes I_{b_1b_2} & \mathbf{0} & (1-a^{[C,f]}_{1,p}(m+1,n))I_{b_1b_2} \end{array} \right]$$

for $n = 1, 2, \ldots, C + f - m - 1$.

Lastly, for levels $m = 1, 2, \ldots, C + f$, the lower diagonal blocks of $Q^{[C,f]}$ are given by

$$Q^{[C,f]}_{m,m-1} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C{+}f{-}m{-}1 \\ C{+}f{-}m \end{array} \begin{array}{c} \begin{array}{ccccccc} 0 & 1 & 2 & \cdots & C{+}f{-}m{-}1 & C{+}f{-}m & C{+}f{-}m{+}1 \end{array} \\ \left[\begin{array}{ccccccc} Q^{[C,f]}_{m,m-1,0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & Q^{[C,f]}_{m,m-1,1} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & Q^{[C,f]}_{m,m-1,2} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q^{[C,f]}_{m,m-1,C+f-m-1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & Q^{[C,f]}_{m,m-1,C+f-m} & \mathbf{0} \end{array}\right] \end{array},$$

where for $m = 0$

$$Q^{[C,f]}_{1,0,0} = \left[\begin{array}{cc} \underline{0}'_{s_1 b_1} & \mathbf{0} \\ \gamma^{[0]}_{10}\underline{B}'_{0,1} & \underline{B}'_{0,1}\underline{\gamma}_{10} \end{array}\right],$$

and

$$Q^{[C,f]}_{1,0,n} = \left[\begin{array}{c} \underline{0}'_{s_1 b_1 b_2}\underline{0}_{(s_2+1)b_2} \\ \underline{\gamma}^{[+0]}_{12} \otimes \underline{B}'_{0,1} \otimes I_{b_2} \\ \underline{0}'_{(s_2+1)b_1 b_2}\underline{0}_{(s_2+1)b_2} \end{array}\right], \; n = 1, 2, \ldots, C + f - 1,$$

while for $m = 2, 3, \ldots, C + f$,

$$Q^{[C,f]}_{m,m-1,0} = \left[\begin{array}{cc} \underline{0}'_{s_1 b_1}\underline{0}_{s_1 b_1} & \mathbf{0} \\ \mathbf{0} & \underline{B}'_{0,1}\underline{\beta}_1 \end{array}\right],$$

and

$$Q^{[C,f]}_{m,m-1,n} = \left[\begin{array}{ccc} \underline{0}'_{s_1 b_1 b_2}\underline{0}_{s_1 b_1 b_2} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (1{-}d^{[C,f]}_2(m{-}1,n))\underline{B}'_{0,1}\underline{\beta}_1 \otimes I_{b_2} & d^{[C,f]}_2(m{-}1,n)\underline{\gamma}^{[+0]}_{12} \otimes \underline{B}'_{0,1}\underline{\beta}_1 \otimes I_{b_2} \\ \underline{0}'_{(s_2+1)b_1 b_2}\underline{0}_{s_1 b_1 b_2} & \mathbf{0} & \mathbf{0} \end{array}\right]$$

for $n = 1, 2, \ldots, C + f - m$.

### 3.3.4 Sojourn Time Distribution

In this subsection, we derive the continuous phase-type representation for the sojourn time (i.e., the time between a machine's failure and when its repairs are complete) distribution of a target machine that suffers a class-1 failure, $\mathcal{S}_1$. Our analysis considers the system at steady state, and hence we require the steady-state probabilities of the maintenance system immediately prior to a class-1 failure. Analogous to Equation (2.2), we have

$$q_{m,n,l,y,y_1,y_2} = \frac{\min\{C, C + f - m - n\}\pi_{m,n,l,y,y_1,y_2}}{\sum_{x_1,x_2,w,z,z_1,z_2} \min\{C, C + f - x_1 - x_2\}\pi_{x_1,x_2,w,z,z_1,z_2}}. \tag{3.7}$$

That is, the probability that the system was in state $(m, n, l, y, y_1, y_2)$ immediately prior to a class-1 failure is the ratio of the steady-state class-1 failure rate from state $(m, n, l, y, y_1, y_2)$ and the total steady-state class-1 failure rate over all states.

Now that we have the distribution of the system before the failure, we must consider how the failure causes the state of the system to change. If the mechanic was previously conducting a switch-in and this failure causes a class-1 switch-in to begin, then the switch-in phase occupied prior to the failure has no bearing on the future development of the system since we track interrupted service times, but not interrupted switch-in times. Thus, let

$$q_{m,n,3,\bullet,y_1,y_2} = \sum_{y=1}^{s_2} q_{m,n,3,y,y_1,y_2}$$

be the total probability that the server was conducting a class-2 switch-in, and define

$$\underline{q}_{0,n,3,\bullet} = (q_{0,n,3,\bullet,0,1}, q_{0,n,3,\bullet,0,2}, \dots, q_{0,n,3,\bullet,0,b_2})$$

and

$$\underline{q}_{m,n,3,\bullet} = (q_{m,n,3,\bullet,1,1}, q_{m,n,3,\bullet,1,2}, \dots, q_{m,n,3,\bullet,1,b_2}, q_{m,n,3,\bullet,2,1}, \dots, q_{m,n,3,\bullet,b_1,b_2}).$$

Similarly, for the case of the queue being empty prior to the failure, let

$$q_{0,0,\bullet,\bullet,\bullet,\bullet} = q_{0,0,0,0,0,0} + \sum_{y=1}^{s_0} q_{0,0,5,y,0,0}.$$

We otherwise group the pre-failure probabilities into the following row vectors. For level $m = 1, 2, \dots, C+f-1$, let

$$\begin{aligned}\underline{q}_{m,0} = (&q_{m,0,1,1,1,0}, \dots, q_{m,0,1,1,b_1,0}, q_{m,0,1,2,1,0}, \dots, q_{m,0,1,s_1,b_1,0},\\ &q_{m,0,2,0,1,0}, \dots, q_{m,0,2,0,b_1,0}),\end{aligned}$$

and for $n = 1, 2, \dots, C+f-m$,

$$\begin{aligned}\underline{q}_{m,n,1} &= (q_{m,n,1,1,1,1}, \dots, q_{m,n,1,1,1,b_2}, q_{m,n,1,1,2,1}, \dots, q_{m,n,1,1,b_1,b_2},\\ &\qquad q_{m,n,1,2,1,1}, \dots, q_{m,n,1,s_1,b_1,b_2}),\\ \underline{q}_{m,n,2} &= (q_{m,n,2,0,1,1}, \dots, q_{m,n,2,0,1,b_2}, q_{m,n,2,0,2,1}, \dots, q_{m,n,2,0,b_1,b_2}),\\ \underline{q}_{m,n,3} &= (q_{m,n,3,1,1,1}, \dots, q_{m,n,3,1,1,b_2}, q_{m,n,3,1,2,1}, \dots, q_{m,0,3,1,b_1,b_2},\\ &\qquad q_{m,n,3,2,1,1}, \dots, q_{m,n,3,s_2,b_1,b_2}),\\ \underline{q}_{m,n,4} &= (q_{m,n,4,0,1,1}, \dots, q_{m,n,4,0,1,b_2}, q_{m,n,4,0,2,1}, \dots, q_{m,n,4,0,b_1,b_2}).\end{aligned}$$

Also, for level 0 and $n = 1, 2, \dots, C+f-1$, let

$$\underline{q}_{0,n,3,y} = (q_{0,n,3,y,0,1}, \dots, q_{0,n,3,y,0,b_2}),\ y = 1, 2, \dots, s_2,$$

and

$$\underline{q}_{0,n,4} = (q_{0,n,4,0,0,1}, \dots, q_{0,n,4,0,0,b_2}).$$

Now, for level $m$, $m = 0, 1, \dots, C+f-1$, define the probability row vector

$$\underline{p}_{m+1} = (\underline{p}_{m+1,0}, \underline{p}_{m+1,1}, \dots, \underline{p}_{m+1,C+f-m-1}).$$

For $m > 0$, $\underline{p}_{m+1,0} = \underline{q}_{m,0}$ and

$$\underline{p}_{m+1,n} = (\underline{p}_{m+1,n,1}, \underline{p}_{m+1,n,2}, \underline{p}_{m+1,n,3}, \underline{p}_{m+1,n,4}),\ n = 1, 2, \dots, C+f-m-1,$$

where

$$
\begin{aligned}
\underline{p}_{m+1,n,1} &= \underline{q}_{m,n,1} + a_1^{[C,f]}(m+1,n)\underline{\gamma}_{21} \otimes \underline{q}_{m,n,3,\bullet} + a_{1,p}^{[C,f]}(m+1,n)\underline{\gamma}_{21} \otimes \underline{q}_{m,n,4},\\
\underline{p}_{m+1,n,2} &= \underline{q}_{m,n,2} + a_1^{[C,f]}(m+1,n)\gamma_{21}^{[0]}\underline{q}_{m,n,3,\bullet} + a_{1,p}^{[C,f]}(m+1,n)\gamma_{21}^{[0]}\underline{q}_{m,n,4},\\
\underline{p}_{m+1,n,3} &= \left(1 - a_1^{[C,f]}(m+1,n)\right)\underline{q}_{m,n,3},\\
\underline{p}_{m+1,n,4} &= \left(1 - a_{1,p}^{[C,f]}(m+1,n)\right)\underline{q}_{m,n,4}.
\end{aligned}
$$

Here, we observe that the initial 'level' of the sojourn time distribution will be increased by the new class-1 machine's presence, which is why the first index of the $\underline{p}$'s are one larger than their component $\underline{q}$'s. Additionally, if the mechanic was already at queue 1 or conducting a class-1 switch-in, then the new failure will not require them to make a decision. However, if a class-2 switch-in or service time was underway, then the failure would cause the mechanic to begin a class-1 switch-in with probability $a_1^{[C,f]}(m+1,n)$ or $a_{1,p}^{[C,f]}(m+1,n)$, respectively (and this switch-in will have a duration of zero with probability $\gamma_{21}^{[0]}$). Note as well that since there was already at least one class-1 machine at queue 1, the service phase of the lead class-1 machine was already determined.

For $m = 0$, in addition to the probability of the failure inducing server movements, we need to initialize the lead class-1 machine's service phase according to the probability vector $\underline{\beta}_1$ since there was an empty queue previous to this failure. Hence, we have

$$
\underline{p}_{1,0} = (q_{0,0,\bullet,\bullet,\bullet,\bullet}\underline{\gamma}_{01} \otimes \underline{\beta}_1, q_{0,0,\bullet,\bullet,\bullet,\bullet}\gamma_{01}^{[0]}\underline{\beta}_1)
$$

and

$$
\underline{p}_{1,n} = (\underline{p}_{1,n,1}, \underline{p}_{1,n,2}, \underline{p}_{1,n,3}, \underline{p}_{1,n,4}),\ n = 1, 2, \ldots, C+f-1,
$$

where

$$
\begin{aligned}
\underline{p}_{1,n,1} &= a_1^{[C,f]}(1,n)\underline{\gamma}_{21} \otimes \underline{\beta}_1 \otimes \underline{q}_{0,n,3,\bullet} + a_{1,p}^{[C,f]}(1,n)\underline{\gamma}_{21} \otimes \underline{\beta}_1 \otimes \underline{q}_{0,n,4},\\
\underline{p}_{1,n,2} &= a_1^{[C,f]}(1,n)\gamma_{21}^{[0]}\underline{\beta}_1 \otimes \underline{q}_{0,n,3,\bullet} + a_{1,p}^{[C,f]}(1,n)\gamma_{21}^{[0]}\underline{\beta}_1 \otimes \underline{q}_{0,n,4},\\
\underline{p}_{1,n,3} &= \left(1 - a_1^{[C,f]}(1,n)\right)(\underline{\beta}_1 \otimes \underline{q}_{0,n,3,1}, \underline{\beta}_1 \otimes \underline{q}_{0,n,3,2}, \ldots, \underline{\beta}_1 \otimes \underline{q}_{0,n,3,s_2}),\\
\underline{p}_{1,n,4} &= \left(1 - a_{1,p}^{[C,f]}(1,n)\right)\underline{\beta}_1 \otimes \underline{q}_{0,n,4}.
\end{aligned}
$$

We can now construct the complete steady-state probability row vector of length

$$
(C+f)\left((s_1+1)b_1 + (s_1+s_2+2)b_1b_2\frac{C+f-1}{2}\right)
$$

describing the state of the system immediately after a class-1 failure, namely

$$
\underline{p} = (\underline{p}_{C+f}, \underline{p}_{C+f-1}, \ldots, \underline{p}_1), \tag{3.8}
$$

which satisfies $\underline{p}\,\underline{e}' = 1$. Before constructing the rate matrix for the machine's sojourn time distribution, we make the following observation. Since we are assuming a FCFS order within each queue, no matter the service policy, a target class-1 machine will never have to wait for the service time of any machines that suffer class-1 failures after their own. However, subsequent class-1 failures may still have an impact on the target machine's sojourn time. The reason for this is twofold. A machine that fails after the target and enters behind them in their queue is a machine that cannot be at risk of entering the opposite queue and potentially receiving service before the target. Also, further class-1 machine failures behind the target may yet influence the mechanic, as the switch-in decision probabilities can be unique for every combination of both (positive) queue lengths.

It then follows that to model the sojourn time, we must track both the position of the target class-1 machine within their queue, as well as the total length of their queue. We achieve this by effectively reducing the number of machines that the system needs to track after every class-1 failure following that of the target, such that the number of reductions is the excess queue length behind the target. This is where we make use of the QBD block superscripts, $[C, f]$, as it allows us to construct our generator blocks as functions of $C$ and $f$, which otherwise would have simply been treated as constants. Note that by reducing the number of considered machines, we are not necessarily reducing the maximum that may be in use at a given time. Therefore, it is important to reduce $f$ to zero before reducing $C$. Combined with this use of notation, the application of Equations (3.1)-(3.3) ensure that the true queue lengths are used when referencing the switch-in decision probabilities.

The sojourn time's rate matrix can thus be constructed as follows:

$$\mathcal{R}_1 = \begin{array}{c|cccccccccc|}
 & [C,f] & [C,f-1] & [C,f-2] & \cdots & [C,1] & [C,0] & [C-1,0] & \cdots & [2,0] & [1,0] \\
\hline
[C,f] & \tilde{Q}^{[C,f]} & \tilde{Q}_-^{[C,f]} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\
[C,f-1] & \mathbf{0} & \tilde{Q}^{[C,f-1]} & \tilde{Q}_-^{[C,f-1]} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} \\
[C,f-2] & \mathbf{0} & \mathbf{0} & \tilde{Q}^{[C,f-2]} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} \\
\vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\
[C,1] & \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \tilde{Q}^{[C,1]} & \tilde{Q}_-^{[C,1]} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} \\
[C,0] & \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \mathbf{0} & \tilde{Q}^{[C,0]} & \tilde{Q}_-^{[C,0]} & \ddots & \mathbf{0} & \mathbf{0} \\
[C-1,0] & \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} & \tilde{Q}^{[C-1,0]} & \ddots & \mathbf{0} & \mathbf{0} \\
\vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\
[2,0] & \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \tilde{Q}^{[2,0]} & \tilde{Q}_-^{[2,0]} \\
[1,0] & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \tilde{Q}^{[1,0]}
\end{array},$$

where

$$\tilde{Q}^{[D,g]} = \begin{array}{c|cccccc|}
 & D{+}g & D{+}g{-}1 & D{+}g{-}2 & \cdots & 2 & 1 \\
\hline
D{+}g & Q_{D+g,D+g}^{[D,g]} & Q_{D+g,D+g-1}^{[D,g]} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\
D{+}g{-}1 & \mathbf{0} & Q_{D+g-1,D+g-1}^{[D,g]} & Q_{D+g-1,D+g-2}^{[D,g]} & \ddots & \mathbf{0} & \mathbf{0} \\
D{+}g{-}2 & \mathbf{0} & \mathbf{0} & Q_{D+g-2,D+g-2}^{[D,g]} & \ddots & \mathbf{0} & \mathbf{0} \\
\vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\
2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q_{2,2}^{[D,g]} & Q_{2,1}^{[D,g]} \\
1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & Q_{1,1}^{[D,g]}
\end{array},$$

and

$$\tilde{Q}_{-}^{[D,g]} = \left[\begin{array}{c|ccccc}
 & D+g-1 & D+g-2 & \cdots & 2 & 1 \\ \hline
D+g & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\
D+g-1 & Q^{[D,g]}_{D+g-1,D+g} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\
D+g-2 & \mathbf{0} & Q^{[D,g]}_{D+g-2,D+g-1} & \ddots & \mathbf{0} & \mathbf{0} \\
\vdots & \vdots & \ddots & \ddots & \vdots & \vdots \\
2 & \mathbf{0} & \mathbf{0} & \cdots & Q^{[D,g]}_{2,3} & \mathbf{0} \\
1 & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & Q^{[D,g]}_{1,2}
\end{array}\right],$$

such that $\mathcal{R}_1$ is a square matrix of dimension

$$\ell_1 = \frac{(C+f)(C+f+1)}{2}\left((s_1+1)b_1 + (s_1+s_2+2)b_1b_2\frac{C+f-1}{3}\right).$$

If $f=0$, then $\mathcal{R}_1$ is the bottom right quadrant starting with level $[C,0]$ and top-left block $\tilde{Q}^{[C,0]}$. From the above, the absorption rates are contributed from $Q^{[i,j]}_{1,0}$ subblocks, corresponding to possible transitions which would result in the lead machine (in this case, the target) receiving service and exiting the queue.

With the rate matrix in hand, we return to the initial probability row vector, $\underline{p}$. This vector contains probabilities for the system immediately after the target machine's failure, which considers all $C+f$ machines, as only one machine can fail at a time. This of course implies that at the time instant when the target machine enters its queue, there cannot be any other machines queued behind it. Thus, the initial probability vector corresponding to the phase-type distribution having rate matrix $\mathcal{R}_1$ is $\underline{\Phi}_1 = (\underline{p}, \underline{0}, \underline{0}, \ldots, \underline{0})$, and so it holds that $\mathcal{S}_1 \sim \mathrm{PH}_{\ell_1}(\underline{\Phi}_1, \mathcal{R}_1)$.

We conclude this subsection with the following comment. We have so far considered only the sojourn time distribution of a class-1 machine. If we want the distribution of $\mathcal{S}_2$ for a machine that suffers a class-2 failure, the distribution can be obtained via the interchange of exponential failure rates, service and switch-in time distributions, and transposes of switch-in decision probability matrices (e.g., replace $\mathcal{P}^{1S}$ by $\left(\mathcal{P}^{2S}\right)'$ and $\mathcal{P}^{2S}$ by $\left(\mathcal{P}^{1S}\right)'$). Following this, the class-2 sojourn time distribution can be obtained by simply repeating the analysis contained within this section, treating it as the new class 1 (and hence class 1 as the new class 2), and calculating the equivalent $\underline{\Phi}_2$ and $\mathcal{R}_2$.

## 3.4 Results Concerning the Expected Number of Working Machines

### 3.4.1 Limit Theorems

In this subsection, we investigate some behaviours of the expected number of working machines at steady state, defined as

$$\begin{aligned}\mathrm{E}[N_{\mathrm{W}}] &= \mathrm{E}[\min\{C, C+f-X_1-X_2\}] \\ &= \sum_m \sum_n \sum_l \sum_y \sum_{y_1} \sum_{y_2} \min\{C, C+f-m-n\}\pi_{m,n,l,y,y_1,y_2}. \qquad (3.9)\end{aligned}$$

Specifically, we are interested in the impact of $C$ and $f$ on $\mathrm{E}[N_{\mathrm{W}}]$, so for the sake of clarity within the theorems of this subsection, we adjust our notation slightly so that $N_{\mathrm{W}}^{[C,f]} = \min\{C, C+f-X_1^{[C,f]}-X_2^{[C,f]}\}$ and $\pi_{m,n,l,y,y_1,y_2}^{[C,f]}$ denote the number of working machines and steady-state probabilities, respectively, as functions of $C$ and $f$.

Our first theorem demonstrates the effect of reducing the maximum number of working machines by one to begin a maintenance float.

**Theorem 3.1.** *For a system at steady state with $k = 2, 3, \ldots$ total machines, $\mathrm{E}[N_{\mathrm{W}}^{[k,0]}] > \mathrm{E}[N_{\mathrm{W}}^{[k-1,1]}]$.*

*Proof.* Refer to the Appendix.

**Remark 3.1.** At the end of the proof of Theorem 3.1, we show that $\mathrm{E}[N_{\mathrm{W}}^{[k,0]}] = c_k \mathrm{E}[N_{\mathrm{W}}^{[k-1,1]}]$ where $1 < c_k < \frac{k}{k-1}$, $k = 2, 3, \ldots$, so it follows that the negative impact of reducing the maximum number of working machines to begin a maintenance float goes to 0 as $k \to \infty$. Therefore, we observe that

$$\lim_{k\to\infty} \mathrm{E}[N_{\mathrm{W}}^{[k,0]}] = \lim_{k\to\infty} \mathrm{E}[N_{\mathrm{W}}^{[k-1,1]}] = \lim_{k\to\infty} \mathrm{E}[N_{\mathrm{W}}^{[k,1]}],$$

implying that the act of including a maintenance float of size $f = 1$ does not impact the limit of the expected number of working machines in comparison to not using a maintenance float.

To get an idea if this also holds true for larger maintenance floats, in Figure 3.2 we have plotted $\mathrm{E}[N_{\mathrm{W}}]$ against the total number of machines $k$ (minimum 2), for the cases $[k-f, f]$, $f = 0, 1, \ldots, 10$. In this plot, we have used an exhaustive service policy with exponentially distributed service times (Exp) having means 1 and 20 for classes 1 and 2, respectively. Switch-in times between classes are also exponentially distributed with means 1, 0.5, and 1 for classes 0, 1, and 2, respectively. The total failure rate was $\alpha = 0.05$, with $\alpha_1 = 0.9\alpha$ and $\alpha_2 = 0.1\alpha$, so that most jobs were 'small'. This 90:10 split will be used throughout this chapter unless otherwise specified.

From Figure 3.2, we observe that independent of how many machines we divert to the float, as the total number of machines $k$ is increased, all of the $\mathrm{E}[N_{\mathrm{W}}^{[k-f,f]}]$ values converge to a single limit as the distance between vertically adjacent points goes to 0. Additionally, we can see that increasing $f$ for a fixed $C$ increases $\mathrm{E}[N_{\mathrm{W}}]$, but of course cannot increase it past the value of $C$ based on the definition in Equation (3.9). This limit result is not a coincidence, nor unique to the exhaustive service policy, as we state in our next theorem.

**Theorem 3.2.** *For any service policy and fixed maintenance float size of $f = 0, 1, 2, \ldots$ machines, the limit of the number of working machines satisfies*

$$\mathrm{E}[N_{\mathrm{W}}^{[\infty]}] = \lim_{C\to\infty} \mathrm{E}[N_{\mathrm{W}}^{[C,f]}] \leq \frac{-1}{\alpha_1 \underline{\beta}_1 B_1^{-1} \underline{e}' + \alpha_2 \underline{\beta}_2 B_2^{-1} \underline{e}'}. \tag{3.10}$$

*Additionally, if switch-in times between the class-1 and class-2 queues are identically zero (i.e., $\gamma_{ji}^{[0]} = 1 \ \forall \ i, j \in \{1, 2\}$), then the upper bound will surely be reached, i.e.,*

$$\mathrm{E}[N_{\mathrm{W}}^{[\infty]}] = \frac{-1}{\alpha_1 \underline{\beta}_1 B_1^{-1} \underline{e}' + \alpha_2 \underline{\beta}_2 B_2^{-1} \underline{e}'}. \tag{3.11}$$

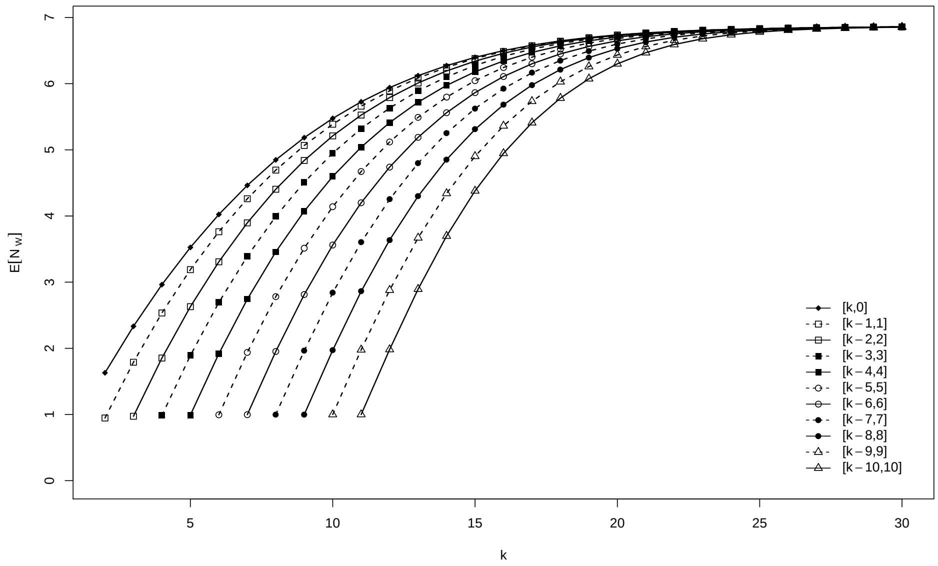

Figure 3.2: Plot of the expected number of working machines $\mathrm{E}[N_{\mathrm{W}}]$ against the total number of machines $k$ for maintenance floats $f = 0, 1, \ldots, 10$, under an exhaustive service policy.

*Proof.* Refer to the Appendix.

**Remark 3.2.** We can re-express Equation (3.11) as

$$\alpha \mathrm{E}[N_{\mathrm{W}}^{[\infty]}] = \mathrm{E}[Z^{\mathrm{M}}]^{-1},$$

where $\alpha \mathrm{E}[N_{\mathrm{W}}^{[\infty]}]$ is the average rate of machine failures as $C \to \infty$ and $\mathrm{E}[Z^{\mathrm{M}}]^{-1}$ is the average rate of machine repairs when the fraction of time that the mechanic is servicing machines goes to 1. Therefore, we can interpret $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}]$ as the expected queue length that reaches an equilibrium which balances the rate of failures with the server's fastest possible rate of repairs. If there are no switch-in times, then any policy can reach this repair rate. However, if switch-ins are possibly incurred when transiting between the class-1 and class-2 queues, then the quantity of these switch-ins (dependent on the service policy) will cause the server to spend a larger fraction of their time idle, lowering their peak repair rate and hence lowering the value of $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}]$ that a policy can reach.

**Remark 3.3.** For a given service policy, if the expected time servicing machines between renewals as defined in the proof of Theorem 3.2, $\mathrm{E}[BP_{\mathrm{ser}}^{[C,f]}]$, increases faster than the expected time switching between queues, $\mathrm{E}[BP_{\mathrm{swi}}^{[C,f]}]$, then by Equation (A.13), the aggregate rate of machine repairs, $\lambda_r^{[C,f]}$, and hence the expected number of working machines, $\mathrm{E}[N_{\mathrm{W}}^{[C,f]}]$, are monotonically increasing in $C$ for a given $f$.

From our numerical analysis, this appears to be normal behaviour, but we were able to replicate a non-monotonic or monotonic decreasing relationship between $\mathrm{E}[N_{\mathrm{W}}]$ and $C$. For example, we observed this in some cases using an unreasonable service policy that sets every decision epoch probability to 1 for both queues (i.e., the mechanic would always switch after observing

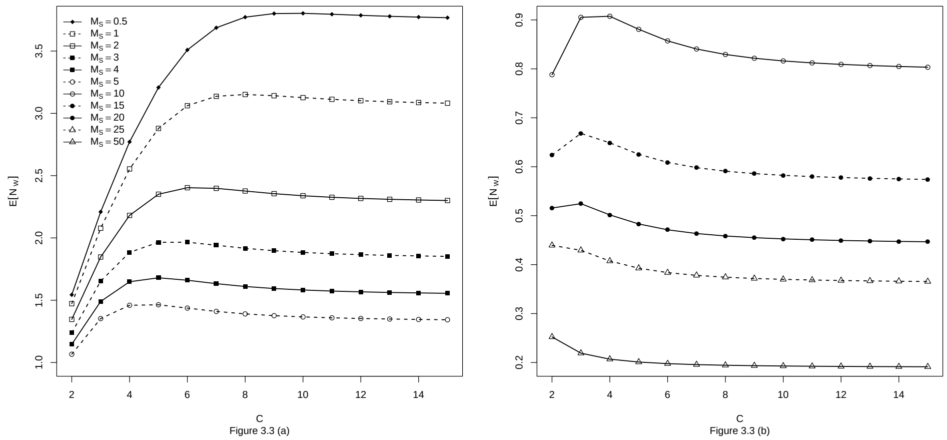

Figure 3.3: Plots of the expected number of working machines $\mathrm{E}[N_{\mathrm{W}}]$ against the capacity $C$ for $f = 0$, $\alpha_1 = \alpha_2 = 0.05$, and exponentially distributed services and switch-in times having means 2 or $M_{\mathrm{S}}$, respectively, under a service policy that maximizes the number of switches.

any arrival to the opposite queue, and after service completions if the opposite queue had a positive length), with the aim of maximizing the number of switches. In Figure 3.3, under this policy, we plot $\mathrm{E}[N_{\mathrm{W}}]$ against $C$ with $f = 0$, symmetric classes having failure rates $\alpha_1 = \alpha_2 = 0.05$ and exponentially distributed service times with mean 2, and exponentially distributed switch-in times for the three classes having equal means of $M_{\mathrm{S}} \in \{0.5, 1, 2, 3, 4, 5, 10, 15, 20, 25, 50\}$. It is clear that a slight non-monotonic relationship is visible in the $M_{\mathrm{S}} = 0.5$ case which becomes more pronounced as $M_{\mathrm{S}}$ increases, eventually turning into a monotonic decreasing relationship in $C$. Omitted from these plots, we also considered the impact of $f$, which had no bearing on the limiting value of $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}]$.

To accompany Theorem 3.2, Table 3.2 presents $\mathrm{E}[N_{\mathrm{W}}^{[C,f]}]$ obtained using the methods in Section 3.3, $\lambda_r^{[C,f]} = \mathrm{E}[N_{\mathrm{W}}^{[C,f]}]/\mathrm{E}[W^{[C,f]}] = \alpha \mathrm{E}[N_{\mathrm{W}}^{[C,f]}]$, simulated values of $\tilde{\mathrm{E}}[BP_{\mathrm{ser}}^{[C,f]}]$ and $\tilde{\mathrm{E}}[BP_{\mathrm{swi}}^{[C,f]}]$ obtained from 500,000 simulated renewal cycles as defined in the proof, as well as the corresponding simulated value

$$\tilde{\lambda}_r^{[C,f]} = \frac{\tilde{\mathrm{E}}[BP_{\mathrm{ser}}^{[C,f]}]/\mathrm{E}[Z^{\mathrm{M}}]}{\frac{1}{C\alpha} + \tilde{\mathrm{E}}[BP_{\mathrm{ser}}^{[C,f]}] + \tilde{\mathrm{E}}[BP_{\mathrm{swi}}^{[C,f]}]}.$$

Select values of $C$ and $f$ are considered, along with several service policies (exhaustive, preemptive resume priority (P), non-preemptive priority (NP), smart Bernoulli (SB), and class-1 $(a, b)$ threshold priority (Thr)). Note that we suppress the superscripts for space considerations. In all cases, the total failure rate was set to $\alpha = 0.05$, and the service times followed hyperexponential-2 ($\mathrm{H}_2$) distributions with initial probability vectors

$$\underline{\beta}_1 = \underline{\beta}_2 = (0.9, 0.1) \tag{3.12}$$

and rate matrices

$$B_1 = 2 \begin{pmatrix} -1 & 0 \\ 0 & -\frac{1}{11} \end{pmatrix}, \; B_2 = \frac{1}{10M_{\mathrm{B}}} \begin{pmatrix} -1 & 0 \\ 0 & -\frac{1}{11} \end{pmatrix}, \tag{3.13}$$

resulting in means of 1 and $20M_{\mathrm{B}}$ for classes 1 and 2, respectively, such that $M_{\mathrm{B}}$ can be used as a scaling factor to adjust the expected size of class-2 jobs, with $M_{\mathrm{B}}$ set to 1 by default. For the switch-in time distributions, we used initial probability vectors

$$\underline{\gamma}_{10} = (p_{>0}, 0),\ \underline{\gamma}_{20} = (0, p_{>0}),\ \underline{\gamma}_{0i} = (0, p_{>0}, 0),\ i = 1, 2, \tag{3.14}$$

and

$$\underline{\gamma}_{ji} = (p_{>0}, 0, 0),\ i, j \in \{1, 2\},\ i \neq j, \tag{3.15}$$

where $p_{>0} = 1 - \gamma_{ji}^{[0]}$ is the probability of a switch-in time being positive in duration, and rate matrices

$$S_1 = \frac{1}{M_{\mathrm{S}}} \begin{pmatrix} -1 & 1 & 0 \\ 0 & -2 & 2 \\ 0 & 0 & -2 \end{pmatrix},\ S_2 = \frac{1}{M_{\mathrm{S}}} \begin{pmatrix} -2 & 2 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & -1 \end{pmatrix}, \tag{3.16}$$

and

$$S_0 = \frac{1}{M_{\mathrm{S}}} \begin{pmatrix} -2 & 0 \\ 0 & -1 \end{pmatrix}, \tag{3.17}$$

where $M_{\mathrm{S}}$ is a scaling factor for all mean switch-in times that is set to 1 by default. These rate matrices imply class-dependent Erlang-2 ($\mathrm{E}_2$) distributed setup times before beginning service, and exponentially distributed take-down times before leaving either class 1 or 2. If the opposite queue is empty and the mechanic would switch to class 0, then they will complete the take-down and only be required to perform a setup after the next failure. As class 1 is being used to denote the smaller jobs, we let these times for class 1 be faster than those for class 2.

For these cases, note that $\mathrm{E}[BP_{\mathrm{swi}}^{[C,f]}] = 0$ when $p_{>0} = 0$, so we omit the corresponding column. In all cases, based on the results of Theorem 3.2, it follows that $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}] \leq 6.896552$ and $\lambda_r^{[\infty]} \leq 0.3448276$. Comparing the $p_{>0} = 0$ and $p_{>0} = 1$ cases, it is clear that the presence of switch-in times reduces the rate at which machines are repaired, as is evident in the values of $\lambda_r^{[C,f]}$.

In the absence of switch-in times, the class-1 preemptive resume priority policy outperforms the others as it prioritizes increasing the expected number of working machines (at the cost of longer class-2 sojourn times) by always choosing to repair small class-1 failures as they occur, to get those machines up and working again as soon as possible. As repaired machines that are put to work are again at risk of failure, a machine that would have otherwise had to wait for a class-2 machine service time to complete could be repaired multiple times during this time span (if it suffers another class-1 failure), effectively increasing the aggregate rate of machine failures.

However, when switch-ins are present, every time a class-1 failure causes the mechanic to leave the class-2 queue, an extra idle period is incurred which reduces the mechanic's efficiency at a noticeable cost to $\lambda_r^{[C,f]}$. In contrast to all other policies, the ratio of $\tilde{\mathrm{E}}[BP_{\mathrm{swi}}^{[C,f]}]$ to $\tilde{\mathrm{E}}[BP_{\mathrm{ser}}^{[C,f]}]$ is by far the highest. The magnitudes of these values for class-1 preemptive priority is due to the fact that the preemptive nature with switch-ins requires a long period of time to actually empty the class-2 queue. With switch-ins, we see the $(5, 5)$ threshold, $(6, 7)$ threshold, and $(9, 9)$ threshold policies maximize $\mathrm{E}[N_{\mathrm{W}}^{[C,f]}]$ for the $[8, 2]$, $[8, 6]$, and $[14, 0]$ cases, respectively. In fact, these are the optimal choices of $a$ and $b$ for $(a, b)$ threshold policies in these positive switch-in cases, as we will demonstrate in Section 3.4.2 for the $[8, 6]$ and $[14, 0]$ cases.

Table 3.2: $\mathrm{E}[N_{\mathrm{W}}^{[C,f]}]$, $\lambda_r^{[C,f]}$, and simulated values of $\mathrm{E}[BP_{\mathrm{ser}}^{[C,f]}]$ and $\mathrm{E}[BP_{\mathrm{swi}}^{[C,f]}]$ for select $C$, $f$, and $p_{>0}$ and various service policies, with $\alpha = 0.05$, $\mathrm{H}_2$ service, and $M_{\mathrm{B}} = M_{\mathrm{S}} = 1$. $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}] \leq 6.896552$ and $\lambda_r^{[\infty]} \leq 0.3448276$.

| | | | | | $p_{>0}$ | | | | |
|---|---|---|---|---|---|---|---|---|---|
| $[C,f]=[8,2]$ | | 0 | | | | | 1 | | |
| Service Policy | $\mathrm{E}[N_{\mathrm{W}}]$ | $\lambda_r$ | $\tilde{\mathrm{E}}[BP_{\mathrm{ser}}]$ | $\tilde{\lambda}_r$ | $\mathrm{E}[N_{\mathrm{W}}]$ | $\lambda_r$ | $\tilde{\mathrm{E}}[BP_{\mathrm{ser}}]$ | $\tilde{\mathrm{E}}[BP_{\mathrm{swi}}]$ | $\tilde{\lambda}_r$ |
| Exhaustive | 5.0419 | 0.2521 | 6.8018 | 0.2521 | 4.8525 | 0.2426 | 10.2967 | 1.8395 | 0.2426 |
| Class-1 P | 5.8966 | 0.2948 | 14.6109 | 0.2944 | 4.0018 | 0.2001 | 206.5702 | 147.1180 | 0.2000 |
| Class-1 NP | 5.1829 | 0.2591 | 7.5561 | 0.2591 | 4.9006 | 0.2450 | 17.6155 | 4.7238 | 0.2445 |
| Class-2 P | 4.9782 | 0.2489 | 6.4578 | 0.2486 | 4.7242 | 0.2362 | 9.4712 | 1.8473 | 0.2363 |
| Class-2 NP | 4.9927 | 0.2496 | 6.5196 | 0.2492 | 4.7503 | 0.2375 | 9.5967 | 1.8281 | 0.2376 |
| (1,0.2) SB | 5.1544 | 0.2577 | 7.3341 | 0.2572 | 4.9016 | 0.2451 | 12.2305 | 2.4778 | 0.2451 |
| (1,0.8) SB | 5.0689 | 0.2534 | 6.9446 | 0.2536 | 4.8657 | 0.2433 | 10.6010 | 1.9495 | 0.2429 |
| (5,5) Thr | 5.5130 | 0.2756 | 9.9895 | 0.2758 | 5.0858 | 0.2543 | 17.6209 | 3.7948 | 0.2541 |
| (6,7) Thr | 5.3217 | 0.2661 | 8.4661 | 0.2662 | 5.0288 | 0.2514 | 13.5382 | 2.5496 | 0.2512 |
| (9,9) Thr | 5.1035 | 0.2552 | 7.1373 | 0.2554 | 4.8971 | 0.2449 | 10.9218 | 1.9408 | 0.2451 |
| $[C,f]=[8,6]$ | | | | | | | | | |
| Exhaustive | 5.3341 | 0.2667 | 8.4668 | 0.2662 | 5.1970 | 0.2599 | 13.7285 | 1.9851 | 0.2599 |
| Class-1 P | 6.3007 | 0.3150 | 26.0904 | 0.3147 | 4.0234 | 0.2012 | 5186.4519 | 3701.8065 | 0.2012 |
| Class-1 NP | 5.5949 | 0.2797 | 10.7111 | 0.2796 | 5.3015 | 0.2651 | 39.8718 | 9.5118 | 0.2650 |
| Class-2 P | 5.2502 | 0.2625 | 8.0464 | 0.2631 | 5.0338 | 0.2517 | 12.1819 | 2.0277 | 0.2514 |
| Class-2 NP | 5.2616 | 0.2631 | 8.0887 | 0.2634 | 5.0591 | 0.2530 | 12.3493 | 1.9994 | 0.2527 |
| (1,0.2) SB | 5.5457 | 0.2773 | 10.2455 | 0.2772 | 5.3380 | 0.2669 | 20.2300 | 3.4138 | 0.2668 |
| (1,0.8) SB | 5.3846 | 0.2692 | 8.8813 | 0.2691 | 5.2331 | 0.2617 | 14.7609 | 2.1864 | 0.2617 |
| (5,5) Thr | 6.1168 | 0.3058 | 19.3409 | 0.3054 | 5.5402 | 0.2770 | 51.5775 | 10.1664 | 0.2768 |
| (6,7) Thr | 5.9823 | 0.2991 | 16.5053 | 0.2995 | 5.5911 | 0.2796 | 33.4503 | 5.3725 | 0.2791 |
| (9,9) Thr | 5.7850 | 0.2892 | 13.0782 | 0.2895 | 5.5203 | 0.2760 | 23.1265 | 3.3062 | 0.2756 |
| $[C,f]=[14,0]$ | | | | | | | | | |
| Exhaustive | 6.1840 | 0.3092 | 12.3103 | 0.3090 | 5.8612 | 0.2931 | 22.7827 | 2.5702 | 0.2933 |
| Class-1 P | 6.7814 | 0.3391 | 83.5556 | 0.3390 | 4.0222 | 0.2011 | 71885.4765 | 51371.5087 | 0.2011 |
| Class-1 NP | 6.3934 | 0.3197 | 18.0328 | 0.3195 | 5.4919 | 0.2746 | 216.4525 | 53.9657 | 0.2746 |
| Class-2 P | 6.0973 | 0.3049 | 10.9145 | 0.3049 | 5.6571 | 0.2829 | 18.4622 | 2.6176 | 0.2828 |
| Class-2 NP | 6.1098 | 0.3055 | 11.0118 | 0.3052 | 5.6904 | 0.2845 | 18.7949 | 2.5705 | 0.2843 |
| (1,0.2) SB | 6.3526 | 0.3176 | 16.6906 | 0.3176 | 5.8373 | 0.2919 | 41.9483 | 6.1635 | 0.2920 |
| (1,0.8) SB | 6.2231 | 0.3112 | 13.2559 | 0.3113 | 5.8672 | 0.2934 | 25.2003 | 2.9786 | 0.2935 |
| (5,5) Thr | 6.6466 | 0.3323 | 38.3886 | 0.3325 | 5.7470 | 0.2874 | 142.2960 | 27.0132 | 0.2874 |
| (6,7) Thr | 6.5632 | 0.3282 | 28.2071 | 0.3282 | 5.9142 | 0.2957 | 73.7993 | 10.8060 | 0.2958 |
| (9,9) Thr | 6.4484 | 0.3224 | 20.5086 | 0.3224 | 5.9709 | 0.2985 | 43.3606 | 5.3009 | 0.2985 |

In Figure 3.2, we observed in the case of the exhaustive service policy that $\mathrm{E}[N_{\mathrm{W}}]$ converged to a limit as we increased the number of machines in the system, a result supported by Theorem 3.2. We now aim to expand on this in Figures 3.4 and 3.5 by plotting $\mathrm{E}[N_{\mathrm{W}}^{[C,f]}]$ for exhaustive, class-$i$ preemptive resume and non-preemptive priority, $i = 1, 2$, as well as $(1, 0.2)$ and $(1, 0.8)$ smart Bernoulli service policies against $C$ for $f = 0$ or $f = 4$. We used the same phase-type service and switch-in distributions used for Table 3.2. As Theorem 3.2 states, the presence of switch-ins will affect the limit of $\mathrm{E}[N_{\mathrm{W}}^{[C,f]}]$, so we consider $p_{>0} = 0$ in Figure 3.4 and $p_{>0} = 1$ in

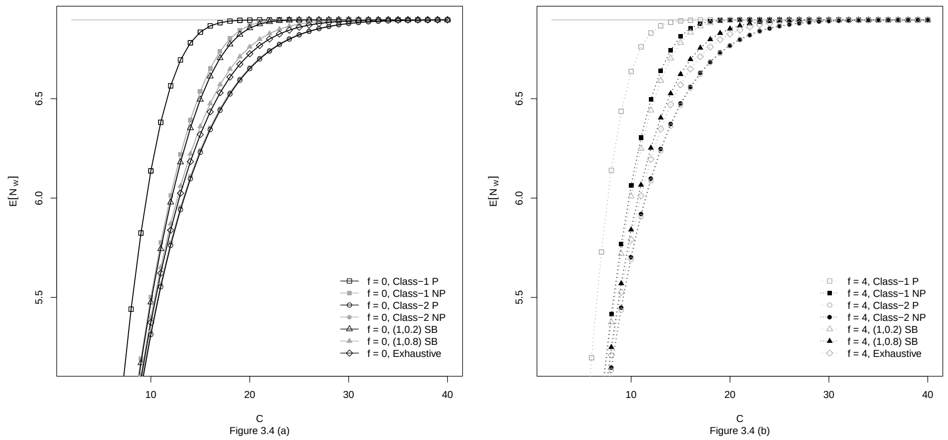

Figure 3.4 (a)

Figure 3.4 (b)

Figure 3.4: Plots of $\mathrm{E}[N_{\mathrm{W}}^{[C,f]}]$ against $C$ for $f = 0, 4$ and select service policies with $\mathrm{H}_2$ service, $p_{>0} = 0$, $M_{\mathrm{B}} = 1$, $M_{\mathrm{S}} = 1$, and $\alpha = 0.05$, where $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}] = 6.896552$.

Figure 3.5. In Figure 3.4, we focus on the $M_{\mathrm{B}} = M_{\mathrm{S}} = 1$ and $\alpha = 0.05$ case, while in Figure 3.5 we also allow $\alpha = 0.1$ and $M_{\mathrm{B}} = 0.5$. Note that due to space constraints, the legend provided in Figure 3.4 is representative of itself as well as Figure 3.5. In all plots, the horizontal grey line is the corresponding limit or upper bound from Equations (3.10) and (3.11).

In Figure 3.4, we confirm that in the absence of switch-in times, this range of service policies all eventually reach the same limiting expected number of working machines, with or without a maintenance float. Unlike Figure 3.2, we are specifically plotting against $C$, and hence the systems plotted in Figure 3.4 (b) have 4 more total machines for a given $C$. An increase in $\mathrm{E}[N_{\mathrm{W}}]$ is observed from the presence of a maintenance float, which increases the speed at which each policy approaches $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}]$. Consistent with Table 3.2, with $p_{>0} = 0$ the preemptive resume priority policy converges to $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}]$ at the highest rate, followed by the other policies in an order depending on their preference to serve class-1 machines (the small jobs) over class-2 machines (the large jobs), with class-2 priority policies performing the worst. Not surprisingly, $(1, 0.2)$ smart Bernoulli is comparable to class-1 non-preemptive priority (which is equivalent to a $(1, 0)$ smart Bernoulli policy), and $(1, 0.8)$ smart Bernoulli is comparable to exhaustive (which is equivalent to a $(1, 1)$ smart Bernoulli policy). There appears to be very little difference between class-2 preemptive resume and non-preemptive priorities, resulting from a combination of low class-2 failure rates relative to class 1 as well as small class-1 service times.

In Figure 3.5, we overlay both the $f = 0$ and $f = 4$ cases on the same plots. In every plot, we observe the same order of service policies in terms of magnitude of $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}]$, with exhaustive having the highest limit (as it incurs the fewest switch-ins) followed by the other policies in reverse order depending on their relative fraction of times spent in a switch-in during a busy period as defined in the proof of Theorem 3.2, i.e., relative to each policy's value of

$$\lim_{C\to\infty} \frac{\mathrm{E}[BP_{\mathrm{swi}}^{[C,f]}]}{\mathrm{E}[BP_{\mathrm{ser}}^{[C,f]}]}.$$

Comparing Figure 5 (a) and (c) to (b) and (d), doubling $\alpha$ approximately halves $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}]$. Comparing the Figure 5 (a) and (b) to (c) and (d), decreasing $M_{\mathrm{B}}$ and hence reducing the size of large jobs increases $\mathrm{E}[Z^{\mathrm{M}}]^{-1}$, increasing the mechanic's peak rate of repair and hence $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}]$. It is also intuitive to observe that the number of machines required to converge to a policy's limiting expected number of working machines depends on the magnitude of $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}]$, and by including a maintenance float without reducing $C$, this limit is reached at a lower value of $C$. For all the plots in Figures 3.4 and 3.5, the convergence to a policy's $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}]$ is monotonic, demonstrating that they satisfy the condition described in Remark 3.3.

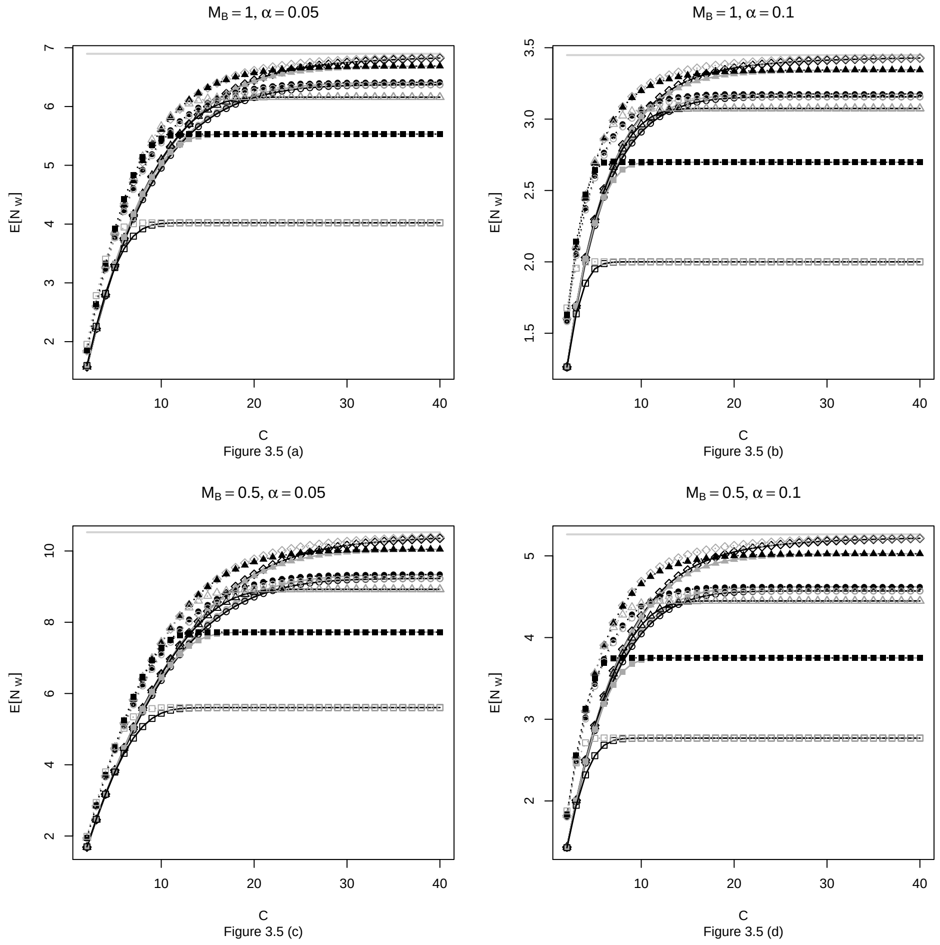

Figure 3.5 (a)

Figure 3.5 (b)

Figure 3.5 (c)

Figure 3.5 (d)

Figure 3.5: Plots of $\mathrm{E}[N_{\mathrm{W}}^{[C,f]}]$ against $C$ for $f = 0, 4$ and select service policies with $\mathrm{H}_2$ service, $p_{>0} = 1$, $M_{\mathrm{B}} = 0.5, 1$, $M_{\mathrm{S}} = 1$, and $\alpha = 0.05, 0.1$.

### 3.4.2 Connection to Mean Sojourn Times

In Section 3.3.4, we showed that the amount of time between a class-1 machine failure and when it is repaired (i.e., its sojourn time) has a $\mathrm{PH}_{\ell_1}(\underline{\Phi}_1, \mathcal{R}_1)$ distribution, and noted that an equivalent $\mathrm{PH}_{\ell_2}(\underline{\Phi}_2, \mathcal{R}_2)$ distribution can be derived for class-2 machines by using the same method after interchanging the class-1 and class-2 failure rates, service and switch-in distributions, and transposes of switch-in decision probability matrices. It then follows that the sojourn time for an arbitrary failed machine, $\mathcal{S}$, is the mixture of $\mathcal{S}_1$ and $\mathcal{S}_2$ having mixing weights equal to the probability of a given failure being of either class.

This is analogous to the simpler model considered in Chapter 2 of this thesis, from which we recall the following results. The PDF of $\mathcal{S}$ is

$$f_{\mathcal{S}}(t) = \frac{\alpha_1}{\alpha}\underline{\Phi}_1 \exp\{\mathcal{R}_1 t\}\underline{\mathcal{R}}'_{0,1} + \frac{\alpha_2}{\alpha}\underline{\Phi}_2 \exp\{\mathcal{R}_2 t\}\underline{\mathcal{R}}'_{0,2},\ t > 0,$$

where $\underline{\mathcal{R}}'_{0,i} = -\mathcal{R}_i\underline{e}'$ is the column vector of absorption rates for the class-$i$ sojourn time distribution, and the $r^{\text{th}}$ moment for $\mathcal{S}$ has formula

$$\mathrm{E}[\mathcal{S}^r] = (-1)^r r!\left(\frac{\alpha_1}{\alpha}\underline{\Phi}_1\mathcal{R}_1^{-r}\underline{e}' + \frac{\alpha_2}{\alpha}\underline{\Phi}_2\mathcal{R}_2^{-r}\underline{e}'\right),$$

implying that $\mathrm{E}[\mathcal{S}] = \frac{\alpha_1}{\alpha}\mathrm{E}[\mathcal{S}_1] + \frac{\alpha_2}{\alpha}\mathrm{E}[\mathcal{S}_2]$. Applying Little's Law [64], we are able to recover the formulas

$$\mathrm{E}[X_i] = \alpha_i \mathrm{E}[N_{\mathrm{W}}]\mathrm{E}[\mathcal{S}_i],\ i = 1, 2,$$

and

$$\mathrm{E}[X_1] + \mathrm{E}[X_2] = \alpha\mathrm{E}[N_{\mathrm{W}}]\mathrm{E}[\mathcal{S}]. \tag{3.18}$$

The advantage of these formulas is that it produces a quicker way to calculate the expected sojourn times, as the expected queue lengths and $\mathrm{E}[N_{\mathrm{W}}]$ only require calculation of the steady-state probabilities and avoids inverting the large rate matrices $\mathcal{R}_1$ and $\mathcal{R}_2$. Equation (3.18) leads us to our third theorem.

**Theorem 3.3.** *For a maintenance system with* $[C, f]$ *machines and a given failure rate* $\alpha$*,* $\mathrm{E}[N_{\mathrm{W}}]$ *will simultaneously be maximized while* $\mathrm{E}[\mathcal{S}]$ *is minimized if* $f = 0$*.*

*Proof.* Recall Equation (3.9), which when $f = 0$ simplifies to

$$\mathrm{E}[N_{\mathrm{W}}] = \mathrm{E}[C - X_1 - X_2] = C - \mathrm{E}[X_1] - \mathrm{E}[X_2]. \tag{3.19}$$

From here, the proof is identical to that of Theorem 2.1, and we recover the non-linear relationship

$$\mathrm{E}[N_{\mathrm{W}}] = \frac{C}{1 + \alpha\mathrm{E}[\mathcal{S}]}. \tag{3.20}$$

It is clear that the selection of a service policy that maximizes $\mathrm{E}[N_{\mathrm{W}}]$ for a given $C$ and $\alpha$ must simultaneously minimize $\mathrm{E}[\mathcal{S}]$.

□

**Remark 3.4.** Equations (3.20) and (A.7) provide an alternate formula for the aggregate rate at which machines fail and are repaired when $f = 0$, namely

$$\lambda_r^{[C,0]} = \frac{C}{\frac{1}{\alpha} + \mathrm{E}[\mathcal{S}^{[C,0]}]}. \tag{3.21}$$

The denominator is equal to the sum of the mean time it takes a working machine to fail and the expected time until it is working again after suffering an arbitrary failure (in a $[C, 0]$ system), and hence is equivalent to the expected time between repairs for a given machine (since there is no maintenance float, a repaired machine is put back to work immediately after it is repaired). The inverse of the time between repairs is the rate of repairs for a single machine, which when multiplied by $C$, results in the aggregate rate of repairs for the entire system.

**Remark 3.5.** If $f \geq 1$, we can obtain an alternative relationship than Equation (3.20) between $\mathrm{E}[N_{\mathrm{W}}]$ and $\mathrm{E}[\mathcal{S}]$. Subtracting Equation (3.18) from $2C + f$ and applying the fact that for any two random variables $X$ and $Y$, $\mathrm{E}[\min\{X, Y\}] + \mathrm{E}[\max\{X, Y\}] = \mathrm{E}[X] + \mathrm{E}[Y]$, we obtain

$$\begin{aligned} 2C + f - \alpha \mathrm{E}[N_{\mathrm{W}}]\mathrm{E}[\mathcal{S}] &= 2C + f - \mathrm{E}[X_1] - \mathrm{E}[X_2] \\ &= \mathrm{E}[N_{\mathrm{W}}] + \mathrm{E}[\max\{C, C + f - X_1 - X_2\}], \end{aligned}$$

which if we rearrange for $\mathrm{E}[N_{\mathrm{W}}]$,

$$\begin{aligned} \mathrm{E}[N_{\mathrm{W}}] &= \frac{2C + f - \mathrm{E}[\max\{C, C + f - X_1 - X_2\}]}{1 + \alpha \mathrm{E}[\mathcal{S}]} \\ &= \frac{C + f - \mathrm{E}[\max\{0, f - X_1 - X_2\}]}{1 + \alpha \mathrm{E}[\mathcal{S}]}, \end{aligned} \tag{3.22}$$

where $\mathrm{E}[\max\{0, f - X_1 - X_2\}]$ is the expected number of functional machines in the maintenance float. Unlike in Equation (3.20) where $\mathrm{E}[N_{\mathrm{W}}]$ and $\mathrm{E}[\mathcal{S}]$ were the only 'variable' components such that one must be maximized when the other is minimized, $\mathrm{E}[\max\{0, f - X_1 - X_2\}]$ will also change if we adjust model parameters or service policies and as such, the simultaneous optimization of $\mathrm{E}[N_{\mathrm{W}}]$ and $\mathrm{E}[\mathcal{S}]$ is not guaranteed.

**Remark 3.6.** From Equations (3.22) and (A.7), we find an alternate equation for the aggregate rate at which machines fail and are repaired to be

$$\lambda_r^{[C,f]} = \frac{C + f - \mathrm{E}[\max\{0, f - X_1^{[C,f]} - X_2^{[C,f]}\}]}{\frac{1}{\alpha} + \mathrm{E}[\mathcal{S}^{[C,f]}]}, \tag{3.23}$$

where the numerator is the expected number of machines in the maintenance system that are in the process of failing (i.e., in use) or the process of being repaired (i.e., are receiving service or are waiting in a queue), and the denominator is the expected amount of time for a machine to fail and then be repaired, agreeing with the intended interpretation of $\lambda_r^{[C,f]}$. Unsurprisingly, Equation (3.23) reduces to Equation (3.21) if we let $f = 0$.

We demonstrate the simultaneous and non-simultaneous optimizations of $\mathrm{E}[N_{\mathrm{W}}]$ and $\mathrm{E}[\mathcal{S}]$ by plotting them over all possible $(a, b)$ threshold policies for the $[8, 6]$ and $[14, 0]$ systems (with switch-ins) considered in Table 3.2. For both figures, grey dashed vertical lines are presented to visually separate the $(a, b)$ threshold policies according to values of $a$. All $(a, b)$ threshold policies are plotted as grey dots by default, while we reuse the symbols from Figures 3.4 and 3.5 for the cases that replicate exhaustive (i.e., $(14, 14)$ threshold) or standard class-1 priority policies (i.e., $(1, 1)$ and $(1, 14)$ threshold). Additionally, the $(a, b)$ threshold policies that maximizes $\mathrm{E}[N_{\mathrm{W}}]$ and/or minimize $\mathrm{E}[\mathcal{S}]$ are plotted as black dots. Finally, black dashed lines are provided for the optimal policies on their corresponding optimal plots for even further visual contrast and to point to their policy on the horizontal axis.

In Figure 3.6, we examine the case of $[C, f] = [8, 6]$, and begin by also plotting the class-1 and class-2 mean sojourn times, $\mathrm{E}[\mathcal{S}_1]$ and $\mathrm{E}[\mathcal{S}_2]$. We observe that the two class-$i$ expected sojourn times have opposite relationships with $a$ and $b$. By increasing the value of $a$ and/or $b$, the strength of the server's preference to serve class 1 before class 2 decreases, as the threshold

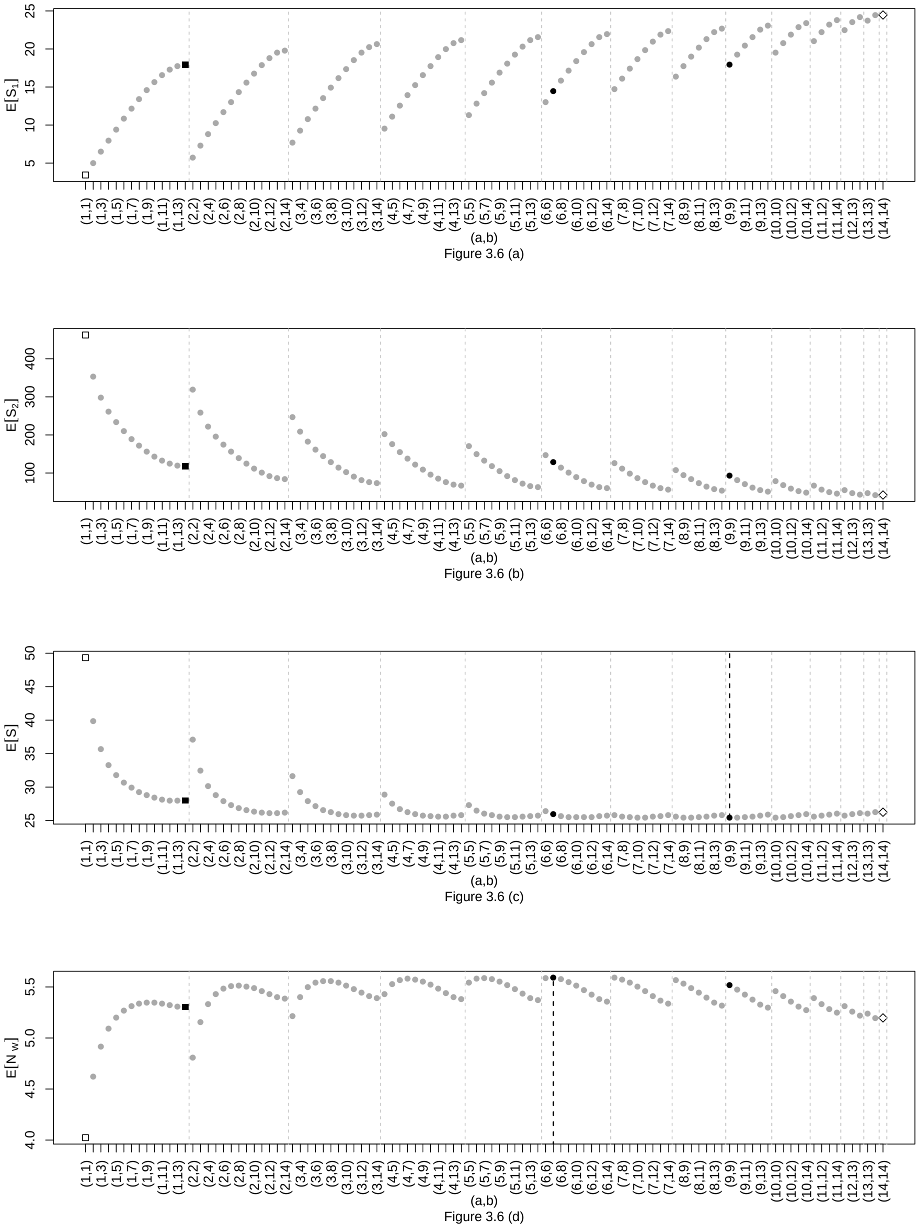

Figure 3.6 (a)

Figure 3.6 (b)

Figure 3.6 (c)

Figure 3.6 (d)

Figure 3.6: Plots of $\mathrm{E}[\mathcal{S}_1]$, $\mathrm{E}[\mathcal{S}_2]$, $\mathrm{E}[\mathcal{S}]$, and $\mathrm{E}[N_{\mathrm{W}}]$ for all possible class-1 $(a, b)$ threshold policies with $[C, f] = [8, 6]$, $\alpha = 0.05$, $\mathrm{H}_2$ service, $p_{>0} = 1$, and $M_{\mathrm{B}} = M_{\mathrm{S}} = 1$.

priorities need larger class-1 queue lengths to activate. Therefore, it follows that increasing $a$ and/or $b$ increases (decreases) $\mathrm{E}[\mathcal{S}_1]$ ($\mathrm{E}[\mathcal{S}_2]$).

As the class-2 expected sojourn times are much larger than the class-1 expected sojourn times, it is not surprising to see in both figures that the overall mean sojourn times are largely decreasing with $a$ and $b$, despite the low 10% mixing weight for class-2 failures, although it begins to increase as a function of $a$ and $b$ for large values of $a$ as the benefit to class 2 for

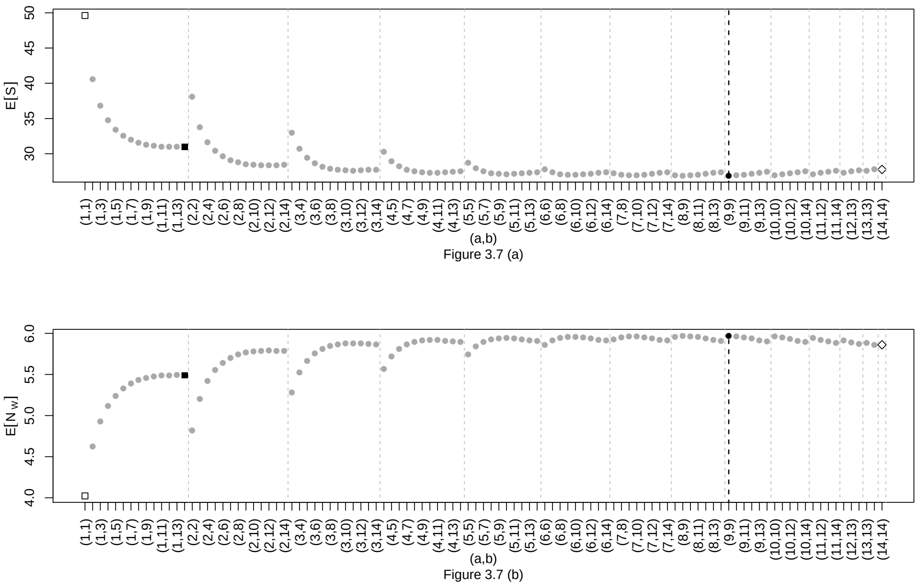

Figure 3.7: Plots of $\mathrm{E}[\mathcal{S}]$ and $\mathrm{E}[N_{\mathrm{W}}]$ for all possible class-1 $(a, b)$ threshold policies with $[C, f] = [14, 0]$, $\alpha = 0.05$, $\mathrm{H}_2$ service, $p_{>0} = 1$, and $M_{\mathrm{B}} = M_{\mathrm{S}} = 1$.

further increasing the thresholds diminishes. In some cases not presented here, it is also possible to see a pronounced concave relationship between $\mathrm{E}[\mathcal{S}]$ and $b$ for small $a$ when there are fewer total machines in the system, but the relation ‘flattens’ for the higher $b$ values as the number of machines (and hence the expected number of working machines) are increased.

The relationship between $\mathrm{E}[N_{\mathrm{W}}]$ and the threshold boundaries is also clearly non-monotonic, as we observe a convex function of $b$ in Figure 3.6 for low values of $a$ before becoming a decreasing function of $b$ for larger $a$’s. This convex relation is ‘flattened’ for high $b$ in Figure 3.7 (similar to the expected sojourn times as mentioned above) as we increase $C$ at the cost of $f$ which results in a net increase in $\mathrm{E}[N_{\mathrm{W}}]$. As these are cases with switch-ins, $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}]$ is increasing in $a$ and $b$ since increasing the thresholds reduces extra switch-ins. Therefore, the decreasing relationship between $\mathrm{E}[N_{\mathrm{W}}]$ and the thresholds for certain ranges of $a$ and $b$ is much less prominent in Figure 3.7 than Figure 3.6, as purely having working machines with no float results in the $[14, 0]$ cases being closer to their limit. In fact, this relationship should approach monotonic increasing in $a$ and $b$ as $C \to \infty$, and the exhaustive policy becomes optimal having the fewest possible switch-ins and hence the highest $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}]$.

Finally, agreeing with the result of Theorem 3.3, we observe simultaneous optimization in the $[14, 0]$ case at $(a, b) = (9, 9)$, resulting in $\mathrm{E}[N_{\mathrm{W}}] = 5.9709$ and $\mathrm{E}[\mathcal{S}] = 26.8941$. Also, the $[8, 6]$ case demonstrates Remark 3.5, where $\mathrm{E}[N_{\mathrm{W}}]$ is maximized at $(a, b) = (6, 7)$ resulting in $\mathrm{E}[N_{\mathrm{W}}] = 5.5911$ and $\mathrm{E}[\mathcal{S}] = 25.9373$, and $\mathrm{E}[\mathcal{S}]$ is minimized at $(9, 9)$ where $\mathrm{E}[N_{\mathrm{W}}] = 5.5203$ and $\mathrm{E}[\mathcal{S}] = 25.4357$.

## 3.5 Numerical Examples

### 3.5.1 $(a, b)$ Threshold Optimization

We now imagine a factory setting where an array of identical machines represent an important component of their production process. To avoid creating a production bottleneck at this step, it is of interest to maximize the average rate at which work is processed by maximizing the expected number of working machines. From Theorem 3.2, we know that there exists a limit $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}]$ dependent on the failure rates and mean service times, which can only be reached if there are no switch-in times. If cost was no object, then this limit could be reached using any service policy given an arbitrarily large $C$ if there were no switch-in times (in fact, it would be advantageous to use class-1 preemptive resume priority which we have seen will reach $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}]$ at the smallest value of $C$). If there are switch-in times corresponding to set-up times for one or both classes, then the exhaustive service policy will have the highest peak service rate and hence the maximum $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}]$.

Unfortunately, increasing your number of machines would have a real cost related to initial investment (e.g., purchase price), recurring costs (e.g., fuel, replacement parts, operational staff), space constraints (e.g., storage space for spares, space on the factory floor for operational machines), and so on. Due to these costs, it may be optimal to invest in a $C$ and $f$ which do not reach the highest possible rate of output. If this is the case, a different policy than exhaustive may be optimal as they have different rates of convergence to the server's peak repair rate and hence could have a higher $\mathrm{E}[N_{\mathrm{W}}^{[C,f]}]$ at a given $C$ and $f$ as seen in Figures 3.6 and 3.7.

With this motivation in mind, we introduce a basic cost function $\mathrm{E}[N_{\mathrm{W}}^{[C,f]}] - r_C C - r_f f$, where $r_C$ is the cost to purchase a machine and to increase the maximum capacity of working machines by one and $r_f$ is the cost per additional machine purchased as a spare and the corresponding cost of storage. Here, we assume that $r_C$ and $r_f$ are normalized with respect to the profit per unit time that a working machine produces, so that maximizing the cost function maximizes the average profit per unit time. We aim to optimize with respect to $C$, $f$, and all possible class-1 $(a, b)$ threshold policies for a given number of machines (i.e., $1 \leq a \leq b \leq C + f$).

For the purposes of our example, we consider a factory with space for a total of $C + f = 14$ machines. We allow $\alpha \in \{0.05, 0.075, 0.10\}$, $M_{\mathrm{B}} \in \{0.5, 1\}$, $M_{\mathrm{S}} \in \{1, 2\}$, and $p_{>0} \in \{0, 0.5, 1\}$, while we assume that the switch-in distributions are of the kind defined in Equations (3.14)-(3.17) in Section 3.4.1. Along with the $\mathrm{H}_2$ service time distributions outlined in Equations (3.12) and (3.13), we also consider Erlang-3 ($\mathrm{E}_3$) distributions having initial probability row vectors

$$\underline{\beta}_1 = \underline{\beta}_2 = (1, 0, 0), \tag{3.24}$$

and rate matrices

$$B_1 = \begin{pmatrix} -3 & 3 & 0 \\ 0 & -3 & 3 \\ 0 & 0 & -3 \end{pmatrix}, \; B_2 = \frac{1}{20M_{\mathrm{B}}} \begin{pmatrix} -3 & 3 & 0 \\ 0 & -3 & 3 \\ 0 & 0 & -3 \end{pmatrix}, \tag{3.25}$$

resulting in the same means as the $\mathrm{H}_2$ service time distributions and maintaining the same interpretation of $M_{\mathrm{B}}$. The $\mathrm{E}_3$ distributions act as good examples of distributions which may be preempted and have a residual service time after the server's return that is less than if they had to restart their work.

Table 3.3: Optimal $C$, $f$, $a$, and $b$, under $\mathrm{H}_2$ and $\mathrm{E}_3$ service for equal machine costs, $(r_C, r_f) = (0.10, 0.10)$.

| | | | | | | | $p_{>0}$ | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $\mathrm{H}_2$ service | | | | 0 | | | 0.5 | | | 1 | |
| $M_\mathrm{B}$ | $M_\mathrm{S}$ | $\alpha$ | $[C,f]$ | $(a,b)$ | $\mathrm{E}[N_\mathrm{W}]$ | $[C,f]$ | $(a,b)$ | $\mathrm{E}[N_\mathrm{W}]$ | $[C,f]$ | $(a,b)$ | $\mathrm{E}[N_\mathrm{W}]$ |
| 1 | 1 | 0.05 | $[13,0]$ | $(1,1)$ | 6.6948 | $[14,0]$ | $(7,7)$ | 6.2251 | $[14,0]$ | $(9,9)$ | 5.9709 |
| | | 0.075 | $[10,0]$ | $(1,1)$ | 4.4827 | $[11,0]$ | $(6,7)$ | 4.1487 | $[11,0]$ | $(8,8)$ | 3.9681 |
| | | 0.10 | $[8,0]$ | $(1,1)$ | 3.3439 | $[8,0]$ | $(5,5)$ | 2.9854 | $[9,0]$ | $(8,8)$ | 2.9386 |
| | 2 | 0.05 | $[13,0]$ | $(1,1)$ | 6.6948 | $[14,0]$ | $(9,9)$ | 5.9549 | $[14,0]$ | $(11,11)$ | 5.5953 |
| | | 0.075 | $[10,0]$ | $(1,1)$ | 4.4827 | $[11,0]$ | $(8,8)$ | 3.9548 | $[12,0]$ | $(11,12)$ | 3.8182 |
| | | 0.10 | $[8,0]$ | $(1,1)$ | 3.3439 | $[9,0]$ | $(7,7)$ | 2.9251 | $[9,0]$ | $(8,9)$ | 2.7411 |
| 0.5 | 1 | 0.05 | $[14,0]$ | $(1,1)$ | 9.4562 | $[14,0]$ | $(5,6)$ | 8.5465 | $[14,0]$ | $(8,8)$ | 8.1463 |
| | | 0.075 | $[13,0]$ | $(1,1)$ | 6.8193 | $[14,0]$ | $(9,9)$ | 6.3171 | $[14,0]$ | $(11,11)$ | 6.0460 |
| | | 0.10 | $[11,0]$ | $(1,1)$ | 5.1484 | $[12,0]$ | $(9,9)$ | 4.7611 | $[13,0]$ | $(12,12)$ | 4.6616 |
| | 2 | 0.05 | $[14,0]$ | $(1,1)$ | 9.4562 | $[14,0]$ | $(8,8)$ | 8.0974 | $[14,0]$ | $(11,11)$ | 7.5325 |
| | | 0.075 | $[13,0]$ | $(1,1)$ | 6.8193 | $[14,0]$ | $(11,11)$ | 6.0146 | $[14,0]$ | $(13,14)$ | 5.6166 |
| | | 0.10 | $[11,0]$ | $(1,1)$ | 5.1484 | $[13,0]$ | $(12,12)$ | 4.6391 | $[14,0]$ | $(13,14)$ | 4.4440 |
| $\mathrm{E}_3$ service | | | | | | | | | | | |
| 1 | 1 | 0.05 | $[11,0]$ | $(1,1)$ | 6.7993 | $[13,0]$ | $(7,8)$ | 6.3602 | $[14,0]$ | $(10,11)$ | 6.1970 |
| | | 0.075 | $[8,0]$ | $(1,1)$ | 4.5144 | $[10,0]$ | $(7,7)$ | 4.2129 | $[11,0]$ | $(9,10)$ | 4.1143 |
| | | 0.10 | $[6,0]$ | $(1,1)$ | 3.3179 | $[8,0]$ | $(5,6)$ | 3.1112 | $[9,0]$ | $(8,9)$ | 3.0543 |
| | 2 | 0.05 | $[11,0]$ | $(1,1)$ | 6.7993 | $[14,0]$ | $(10,10)$ | 6.1884 | $[14,0]$ | $(12,14)$ | 5.8122 |
| | | 0.075 | $[8,0]$ | $(1,1)$ | 4.5144 | $[11,0]$ | $(9,10)$ | 4.1065 | $[12,0]$ | $(11,12)$ | 3.9632 |
| | | 0.10 | $[6,0]$ | $(1,1)$ | 3.3179 | $[9,0]$ | $(8,9)$ | 3.0466 | $[9,0]$ | $(8,9)$ | 2.8523 |
| 0.5 | 1 | 0.05 | $[14,0]$ | $(1,1)$ | 10.2076 | $[14,0]$ | $(5,7)$ | 9.0983 | $[14,0]$ | $(8,10)$ | 8.6314 |
| | | 0.075 | $[11,0]$ | $(1,1)$ | 6.9260 | $[14,0]$ | $(11,14)$ | 6.6001 | $[14,0]$ | $(13,14)$ | 6.3492 |
| | | 0.10 | $[9,0]$ | $(1,1)$ | 5.2021 | $[11,0]$ | $(9,11)$ | 4.8821 | $[12,0]$ | $(11,12)$ | 4.7810 |
| | 2 | 0.05 | $[14,0]$ | $(1,1)$ | 10.2076 | $[14,0]$ | $(7,11)$ | 8.5985 | $[14,0]$ | $(12,14)$ | 7.9602 |
| | | 0.075 | $[11,0]$ | $(1,1)$ | 6.9260 | $[14,0]$ | $(13,14)$ | 6.3362 | $[14,0]$ | $(13,14)$ | 5.9164 |
| | | 0.10 | $[9,0]$ | $(1,1)$ | 5.2021 | $[12,0]$ | $(11,12)$ | 4.7730 | $[14,0]$ | $(13,14)$ | 4.6623 |

We begin by considering optimization when $r_C = r_f$ (i.e., when every machine costs the same whether it will increase the system's capacity or act as a spare in the maintenance float). Table 3.3 contains the optimal $[C, f]$, $(a, b)$, and $\mathrm{E}[N_\mathrm{W}^{[C,f]}]$ (with suppressed superscripts) at $r_C = r_f = 0.10$ over the combinations of parameters and service time distributions outlined above. The first observation that stands out is that in no cases is it optimal to have $f \geq 1$, even when $C < 14$ allows for more additional machines before hitting the cap. Recalling Theorem 3.1, we have that $\mathrm{E}[N_\mathrm{W}^{[k,0]}] > \mathrm{E}[N_\mathrm{W}^{[k-1,1]}]$. We have also seen that for the exhaustive service policy example in Figure 3.2, $\mathrm{E}[N_\mathrm{W}^{[k-f,f]}]$ is a decreasing function of $f$ (although $\mathrm{E}[N_\mathrm{W}^{[C,f]}]$ is typically an increasing function of $f$). Therefore, it makes sense in this case to never invest in a maintenance float if putting all machines towards $C$ maximizes $\mathrm{E}[N_\mathrm{W}^{[C,f]}]$ for a given $C + f$, when $r_C = r_f$ reduces the cost function to $\mathrm{E}[N_\mathrm{W}^{[C,f]}] - r_C(C + f)$. Thus, it is only ever of financial interest to invest in a maintenance float if it is cheaper to add a spare machine to the system than it is to increase $C$ (i.e., $r_C > r_f$). This brings us to our next result.

**Theorem 3.4** *Under cost function* $\mathrm{E}[N_{\mathrm{W}}] - r_C C - r_f f$*, if* $r_C > r_f$*, then for a system with* $k$ *total machines,* $k = 2, 3, \ldots$*, it will be suboptimal to not use a maintenance float if*

$$\mathrm{E}[N_{\mathrm{W}}^{[k,0]}] < k(r_C - r_f). \tag{3.26}$$

*Proof.* Recall from the proof of Theorem 3.1 that $\mathrm{E}[N_{\mathrm{W}}^{[k,0]}] = c_k \mathrm{E}[N_{\mathrm{W}}^{[k-1,1]}]$, where $1 < c_k < \frac{k}{k-1}$, $k = 2, 3, \ldots$. It will be suboptimal to select $f = 0$ in a system having $k$ total machines if

$$\mathrm{E}[N_{\mathrm{W}}^{[k-1,1]}] - (k-1)r_C - r_f > \mathrm{E}[N_{\mathrm{W}}^{[k,0]}] - kr_C,$$

or equivalently,

$$\mathrm{E}[N_{\mathrm{W}}^{[k,0]}] - \mathrm{E}[N_{\mathrm{W}}^{[k-1,1]}] < r_C - r_f.$$

We observe that

$$\mathrm{E}[N_{\mathrm{W}}^{[k,0]}] - \mathrm{E}[N_{\mathrm{W}}^{[k-1,1]}] = \mathrm{E}[N_{\mathrm{W}}^{[k,0]}](1 - c_k^{-1}) < \mathrm{E}[N_{\mathrm{W}}^{[k,0]}]\left(1 - \frac{k-1}{k}\right) = \frac{1}{k}\mathrm{E}[N_{\mathrm{W}}^{[k,0]}].$$

Thus, if Equation (3.26) holds, then

$$r_C - r_f > \frac{1}{k}\mathrm{E}[N_{\mathrm{W}}^{[k,0]}] > \mathrm{E}[N_{\mathrm{W}}^{[k,0]}] - \mathrm{E}[N_{\mathrm{W}}^{[k-1,1]}],$$

and so it would be suboptimal to select $f = 0$ under the given cost function.

□

Note that it may be optimal to use a float when the inequality of Theorem 3.4 does not hold (i.e., for small $k$), so long as $r_C > r_f$ is still true. Theorem 3.4 simply provides an inequality that, if it holds, guarantees that $f = 0$ is not optimal at $k$ total machines. By Theorem 3.2, we know that $\lim_{C\to\infty} \mathrm{E}[N_{\mathrm{W}}^{[C,0]}]$ has a finite upper bound, which implies that there must exist a $k \in \mathbb{Z}^+$ such that Equation (3.26) is satisfied if $r_C > r_f$. Additionally, this implies that if we increase $r_C$ for a given $k \geq 2$ and $r_f$, we will eventually reach a point for sure where it becomes optimal to use a maintenance float.

Other conclusions also follow from Table 3.3. In particular, when $p_{>0} = 0$, it is optimal to use class-1 preemptive resume priority, as it reaches its $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}]$ at the smallest value of $C$ out of the considered policies, and this limit is not penalized by its additional switches due to identically zero switch-in times. We observe that some optimal $C$ are less than 14, corresponding to situations where there is no $(a, b)$ which would result in $\mathrm{E}[N_{\mathrm{W}}^{[C+1,0]}]$ that is at least $r_C = 0.10$ greater than the optimal $\mathrm{E}[N_{\mathrm{W}}^{[C,0]}]$. In fact, the $\mathrm{E}_3$ distributions in contrast to $\mathrm{H}_2$ often results in a smaller optimal $C$ while simultaneously allowing a larger $\mathrm{E}[N_{\mathrm{W}}^{[C,0]}]$, an advantage of having a lower service time variance and a type of partitioned sequential work which benefits more from the nature of the preemptive resume priorities (whereas $\mathrm{H}_2$ simply remembers which exponential distribution from the mixture the job belonged to).

Decreasing $M_{\mathrm{B}}$ results in faster class-2 services, thereby increasing $\mathrm{E}[Z^{\mathrm{M}}]^{-1}$ (and hence $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}]$). The higher limit requires more total machines to reach it, and hence increments in optimal $\mathrm{E}[N_{\mathrm{W}}^{[C,0]}]$ (i.e., at optimal $(a, b)$ threshold cases for given $C$) will outweigh the cost of an additional machine until larger values of $C$. This results in larger optimal $C$ and $\mathrm{E}[N_{\mathrm{W}}^{[C,0]}]$. Increasing $\alpha$ has the opposite effect on $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}]$, resulting in a lower limit and hence lower optimal $C$ values. Increasing $M_{\mathrm{S}}$ for a given positive $p_{>0}$ or increasing $p_{>0}$ for a given $M_{\mathrm{S}}$ (or increasing both) causes switch-ins to be more costly, penalizing a priority policy inversely proportional to $a$ and $b$, while lowering $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}]$. This has the effect of increasing optimal threshold limits while lowering the optimal $\mathrm{E}[N_{\mathrm{W}}^{[C,0]}]$.

Next, we allow increasing the number of float machines to cost half as much as increasing the system capacity (i.e., $r_f = r_C/2$) and consider a range of costs $r_C \in \{0.05, 0.10, 0.25\}$ for $\mathrm{H}_2$ service in Table 3.4 and $\mathrm{E}_3$ service in Table 3.5. Comparing the $r_C = 0.10$ cases from these two tables to Table 3.3, it is possible (for the cases that did not already select $C = 14$) to increase the maintenance float by 2 machines for the cost of 1 capacity slot, which allows those cases to 'afford' to increase the total number of machines. While a single increase in $C$ is worth more than a single increase in $f$, the benefit of multiple spares can outweigh that of a single capacity machine when $C$ is already larger than $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}]$. This follows since the fraction of time that the system spends near capacity decreases as $C$ becomes large and so the marginal benefit of increasing $C$ over $f$ will reduce, and increasing $f$ multiple times can outweigh a unit increase of $C$ when it is larger than $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}]$ (recall that increasing $f$ cannot result in $N_{\mathrm{W}}$ surpassing $C$, and so the benefit of increasing $C$ over $f$ is much larger for $C < \mathrm{E}[N_{\mathrm{W}}^{[\infty]}]$). We also observe some cases where $C = 14$ in Table 3.3 but some capacity is diverted to the float, slightly decreasing $\mathrm{E}[N_{\mathrm{W}}^{[C,f]}]$ but saving much more in costs.

Our earlier observations concerning parameters $M_{\mathrm{B}}$, $M_{\mathrm{S}}$, $\alpha$, and $p_{>0}$ clearly still hold true in Tables 3.4 and 3.5. Also, we still observe $\mathrm{E}_3$ service achieving higher $\mathrm{E}[N_{\mathrm{W}}^{[C,f]}]$ while typically selecting optimal $C$ that are no larger than those selected for $\mathrm{H}_2$ service, with the exception of the $r_C = 0.25$, $M_{\mathrm{B}} = 0.5$, $M_{\mathrm{S}} = 2$, $\alpha = 0.10$ case where the faster rate at which $\mathrm{E}[N_{\mathrm{W}}^{[C,f]}]$ approaches its limit for $\mathrm{E}_3$ allows it to 'afford' to increase $C$ longer than $\mathrm{H}_2$ service. Finally, we observe that by increasing the cost per machine, the system will want to optimize at fewer machines as the incremental costs will begin to outpace the increases in $\mathrm{E}[N_{\mathrm{W}}^{[C,f]}]$ at fewer machines. When optimizing the $(a, b)$ threshold at fewer machines, the decreases in peak repair rate caused by extra incurred switch-ins are smaller due to fewer observed failures, and so the optimal $a$ and $b$ are non-increasing in $r_C$ (and $r_f$).

### 3.5.2 Class-1 Sojourn Time Densities for $(a, a)$ Threshold Policies

In Section 3.3.4, we derived the distribution for a class-1 machine's sojourn time, $\mathcal{S}_1$, to be $\mathrm{PH}_{\ell_1}(\underline{\Phi}_1, \mathcal{R}_1)$, resulting in the PDF

$$f_{\mathcal{S}_1}(t) = \underline{\Phi}_1 \exp\{\mathcal{R}_1 t\} \underline{R}'_{0,1}. \tag{3.27}$$

As an illustration, we plot some of these densities for a family of service policies. For the sake of brevity, we constrain ourselves to the set of $(a, a)$ threshold policies (i.e., preemptive resume threshold policies), which exhibited notable sensitivities to the selection of threshold parameter $a$, much more so than the $(a, C+f)$ threshold policies (i.e., non-preemptive threshold policies) in the numerical cases we considered. As computing the matrix exponential function in Equation (3.27) can be quite time consuming for systems with large state spaces, we consider only Exp service time distributions within this example (having the typical means of 1 and $20M_{\mathrm{B}}$ for class 1 and class 2, respectively), along with the modest number of machines $C = 8$ and $f = 2$.

Table 3.4: Optimal $C$, $f$, $a$, and $b$, under $H_2$ service and cheaper reserve machines ($r_f = r_C/2$).

| | | | | | | $p_{>0}$ | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $\underline{r} = (0.05, 0.025)$ | | | | 0 | | | 0.5 | | | 1 | |
| $M_B$ | $M_S$ | $\alpha$ | $[C, f]$ | $(a, b)$ | $E[N_W]$ | $[C, f]$ | $(a, b)$ | $E[N_W]$ | $[C, f]$ | $(a, b)$ | $E[N_W]$ |
| 1 | 1 | 0.05 | $[12, 2]$ | $(1, 1)$ | 6.7511 | $[13, 1]$ | $(7, 7)$ | 6.2110 | $[13, 1]$ | $(9, 9)$ | 5.9579 |
| | | 0.075 | $[9, 3]$ | $(1, 1)$ | 4.5526 | $[9, 5]$ | $(8, 8)$ | 4.2851 | $[10, 4]$ | $(11, 11)$ | 4.1606 |
| | | 0.10 | $[7, 3]$ | $(1, 1)$ | 3.4093 | $[7, 7]$ | $(9, 9)$ | 3.2572 | $[8, 6]$ | $(11, 12)$ | 3.1780 |
| | 2 | 0.05 | $[12, 2]$ | $(1, 1)$ | 6.7511 | $[13, 1]$ | $(9, 9)$ | 5.9411 | $[13, 1]$ | $(11, 11)$ | 5.5850 |
| | | 0.075 | $[9, 3]$ | $(1, 1)$ | 4.5526 | $[10, 4]$ | $(10, 11)$ | 4.1517 | $[11, 3]$ | $(13, 14)$ | 3.9622 |
| | | 0.10 | $[7, 3]$ | $(1, 1)$ | 3.4093 | $[8, 6]$ | $(11, 11)$ | 3.1728 | $[9, 5]$ | $(13, 14)$ | 3.0523 |
| 0.5 | 1 | 0.05 | $[14, 0]$ | $(1, 1)$ | 9.4562 | $[14, 0]$ | $(5, 6)$ | 8.5465 | $[14, 0]$ | $(8, 8)$ | 8.1463 |
| | | 0.075 | $[12, 2]$ | $(1, 1)$ | 6.8780 | $[13, 1]$ | $(9, 9)$ | 6.3040 | $[12, 2]$ | $(11, 11)$ | 6.0114 |
| | | 0.10 | $[9, 4]$ | $(1, 1)$ | 5.1964 | $[11, 3]$ | $(11, 11)$ | 4.8967 | $[11, 3]$ | $(13, 13)$ | 4.7217 |
| | 2 | 0.05 | $[14, 0]$ | $(1, 1)$ | 9.4562 | $[14, 0]$ | $(8, 8)$ | 8.0974 | $[14, 0]$ | $(11, 11)$ | 7.5325 |
| | | 0.075 | $[12, 2]$ | $(1, 1)$ | 6.8780 | $[13, 1]$ | $(11, 11)$ | 6.0035 | $[12, 2]$ | $(13, 14)$ | 5.5904 |
| | | 0.10 | $[9, 4]$ | $(1, 1)$ | 5.1964 | $[11, 3]$ | $(13, 13)$ | 4.7008 | $[11, 3]$ | $(13, 14)$ | 4.4221 |
| $\underline{r} = (0.10, 0.05)$ | | | | | | | | | | | |
| 1 | 1 | 0.05 | $[11, 3]$ | $(1, 1)$ | 6.7098 | $[11, 3]$ | $(6, 6)$ | 6.1403 | $[11, 3]$ | $(8, 8)$ | 5.8850 |
| | | 0.075 | $[8, 3]$ | $(1, 1)$ | 4.4901 | $[8, 5]$ | $(7, 7)$ | 4.2039 | $[8, 6]$ | $(9, 9)$ | 4.0882 |
| | | 0.10 | $[6, 3]$ | $(1, 1)$ | 3.3414 | $[6, 5]$ | $(6, 6)$ | 3.1177 | $[6, 5]$ | $(8, 8)$ | 2.9792 |
| | 2 | 0.05 | $[11, 3]$ | $(1, 1)$ | 6.7098 | $[11, 3]$ | $(8, 8)$ | 5.8708 | $[11, 3]$ | $(11, 11)$ | 5.5148 |
| | | 0.075 | $[8, 3]$ | $(1, 1)$ | 4.4901 | $[8, 6]$ | $(9, 9)$ | 4.0810 | $[9, 5]$ | $(12, 12)$ | 3.8996 |
| | | 0.10 | $[6, 3]$ | $(1, 1)$ | 3.3414 | $[6, 5]$ | $(7, 8)$ | 2.9720 | $[7, 5]$ | $(11, 11)$ | 2.8933 |
| 0.5 | 1 | 0.05 | $[14, 0]$ | $(1, 1)$ | 9.4562 | $[14, 0]$ | $(5, 6)$ | 8.5465 | $[13, 1]$ | $(8, 8)$ | 8.1062 |
| | | 0.075 | $[11, 3]$ | $(1, 1)$ | 6.8373 | $[11, 3]$ | $(8, 8)$ | 6.2307 | $[11, 3]$ | $(11, 11)$ | 5.9649 |
| | | 0.10 | $[9, 3]$ | $(1, 1)$ | 5.1613 | $[9, 5]$ | $(9, 9)$ | 4.8288 | $[9, 5]$ | $(12, 12)$ | 4.6434 |
| | 2 | 0.05 | $[14, 0]$ | $(1, 1)$ | 9.4562 | $[13, 1]$ | $(7, 8)$ | 8.0555 | $[13, 1]$ | $(11, 11)$ | 7.5060 |
| | | 0.075 | $[11, 3]$ | $(1, 1)$ | 6.8373 | $[11, 3]$ | $(10, 10)$ | 5.9331 | $[11, 3]$ | $(13, 13)$ | 5.5533 |
| | | 0.10 | $[9, 3]$ | $(1, 1)$ | 5.1613 | $[9, 5]$ | $(11, 11)$ | 4.6228 | $[9, 5]$ | $(13, 14)$ | 4.3509 |
| $\underline{r} = (0.25, 0.125)$ | | | | | | | | | | | |
| 1 | 1 | 0.05 | $[9, 2]$ | $(1, 1)$ | 6.2249 | $[9, 3]$ | $(3, 5)$ | 5.7651 | $[9, 3]$ | $(6, 6)$ | 5.4992 |
| | | 0.075 | $[6, 2]$ | $(1, 1)$ | 4.0503 | $[6, 2]$ | $(1, 4)$ | 3.6371 | $[6, 2]$ | $(5, 5)$ | 3.4244 |
| | | 0.10 | $[5, 1]$ | $(1, 1)$ | 2.9977 | $[4, 2]$ | $(1, 3)$ | 2.5299 | $[4, 2]$ | $(3, 4)$ | 2.3718 |
| | 2 | 0.05 | $[9, 2]$ | $(1, 1)$ | 6.2249 | $[8, 4]$ | $(4, 6)$ | 5.3665 | $[8, 4]$ | $(7, 8)$ | 5.0121 |
| | | 0.075 | $[6, 2]$ | $(1, 1)$ | 4.0503 | $[6, 2]$ | $(1, 5)$ | 3.4294 | $[5, 3]$ | $(5, 6)$ | 3.0341 |
| | | 0.10 | $[5, 1]$ | $(1, 1)$ | 2.9977 | $[4, 2]$ | $(1, 4)$ | 2.3846 | $[4, 1]$ | $(4, 5)$ | 2.0562 |
| 0.5 | 1 | 0.05 | $[13, 1]$ | $(1, 1)$ | 9.3829 | $[12, 2]$ | $(3, 6)$ | 8.3904 | $[12, 2]$ | $(8, 8)$ | 8.0157 |
| | | 0.075 | $[10, 2]$ | $(1, 1)$ | 6.5878 | $[9, 3]$ | $(4, 6)$ | 5.8294 | $[9, 4]$ | $(8, 9)$ | 5.6796 |
| | | 0.10 | $[7, 2]$ | $(1, 1)$ | 4.7393 | $[7, 2]$ | $(2, 5)$ | 4.2149 | $[7, 3]$ | $(7, 7)$ | 4.1154 |
| | 2 | 0.05 | $[13, 1]$ | $(1, 1)$ | 9.3829 | $[12, 2]$ | $(5, 8)$ | 7.9720 | $[11, 3]$ | $(10, 10)$ | 7.3296 |
| | | 0.075 | $[10, 2]$ | $(1, 1)$ | 6.5878 | $[9, 4]$ | $(6, 9)$ | 5.6500 | $[9, 4]$ | $(11, 11)$ | 5.2731 |
| | | 0.10 | $[7, 2]$ | $(1, 1)$ | 4.7393 | $[6, 3]$ | $(3, 6)$ | 3.8452 | $[6, 3]$ | $(8, 8)$ | 3.5430 |

Table 3.5: Optimal $C$, $f$, $a$, and $b$, under $\mathrm{E}_3$ service and cheaper reserve machines ($r_f = r_C/2$).

| | | | | | | | $p_{>0}$ | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $\underline{r} = (0.05, 0.025)$ | | | | 0 | | | 0.5 | | | 1 | |
| $M_\mathrm{B}$ | $M_\mathrm{S}$ | $\alpha$ | $[C,f]$ | $(a,b)$ | $\mathrm{E}[N_\mathrm{W}]$ | $[C,f]$ | $(a,b)$ | $\mathrm{E}[N_\mathrm{W}]$ | $[C,f]$ | $(a,b)$ | $\mathrm{E}[N_\mathrm{W}]$ |
| 1 | 1 | 0.05 | $[9,4]$ | $(1,1)$ | 6.8564 | $[11,3]$ | $(8,8)$ | 6.4220 | $[12,2]$ | $(10,10)$ | 6.1761 |
| | | 0.075 | $[7,2]$ | $(1,1)$ | 4.5630 | $[8,6]$ | $(10,10)$ | 4.3862 | $[9,5]$ | $(11,12)$ | 4.2653 |
| | | 0.10 | $[5,3]$ | $(1,1)$ | 3.4224 | $[6,6]$ | $(9,9)$ | 3.2713 | $[7,7]$ | $(12,13)$ | 3.2401 |
| | 2 | 0.05 | $[9,4]$ | $(1,1)$ | 6.8564 | $[12,2]$ | $(10,10)$ | 6.1678 | $[12,2]$ | $(11,13)$ | 5.7829 |
| | | 0.075 | $[7,2]$ | $(1,1)$ | 4.5630 | $[9,5]$ | $(11,12)$ | 4.2624 | $[10,4]$ | $(13,14)$ | 4.0810 |
| | | 0.10 | $[5,3]$ | $(1,1)$ | 3.4224 | $[8,6]$ | $(12,13)$ | 3.2635 | $[9,5]$ | $(13,14)$ | 3.1579 |
| 0.5 | 1 | 0.05 | $[13,1]$ | $(1,1)$ | 10.1838 | $[14,0]$ | $(5,7)$ | 9.0983 | $[14,0]$ | $(8,10)$ | 8.6314 |
| | | 0.075 | $[9,4]$ | $(1,1)$ | 6.9852 | $[12,2]$ | $(10,13)$ | 6.5812 | $[12,2]$ | $(13,14)$ | 6.3296 |
| | | 0.10 | $[7,4]$ | $(1,1)$ | 5.2438 | $[10,4]$ | $(13,14)$ | 5.0609 | $[10,4]$ | $(13,14)$ | 4.9010 |
| | 2 | 0.05 | $[13,1]$ | $(1,1)$ | 10.1838 | $[14,0]$ | $(7,11)$ | 8.5985 | $[13,1]$ | $(12,14)$ | 7.9388 |
| | | 0.075 | $[9,4]$ | $(1,1)$ | 6.9852 | $[12,2]$ | $(13,14)$ | 6.3152 | $[12,2]$ | $(13,14)$ | 5.9028 |
| | | 0.10 | $[7,4]$ | $(1,1)$ | 5.2438 | $[10,4]$ | $(13,14)$ | 4.9054 | $[10,4]$ | $(13,14)$ | 4.6350 |
| $\underline{r} = (0.10, 0.05)$ | | | | | | | | | | | |
| 1 | 1 | 0.05 | $[9,3]$ | $(1,1)$ | 6.8209 | $[9,5]$ | $(7,7)$ | 6.3457 | $[10,4]$ | $(9,9)$ | 6.1118 |
| | | 0.075 | $[6,3]$ | $(1,1)$ | 4.5221 | $[6,6]$ | $(7,7)$ | 4.2330 | $[7,6]$ | $(9,10)$ | 4.1480 |
| | | 0.10 | $[5,2]$ | $(1,1)$ | 3.3846 | $[5,5]$ | $(6,6)$ | 3.1620 | $[6,5]$ | $(9,9)$ | 3.1047 |
| | 2 | 0.05 | $[9,3]$ | $(1,1)$ | 6.8209 | $[10,4]$ | $(9,9)$ | 6.1043 | $[11,3]$ | $(11,12)$ | 5.7471 |
| | | 0.075 | $[6,3]$ | $(1,1)$ | 4.5221 | $[7,6]$ | $(9,10)$ | 4.1440 | $[8,6]$ | $(12,13)$ | 3.9933 |
| | | 0.10 | $[5,2]$ | $(1,1)$ | 3.3846 | $[6,5]$ | $(8,9)$ | 3.1012 | $[7,4]$ | $(10,11)$ | 2.9589 |
| 0.5 | 1 | 0.05 | $[13,1]$ | $(1,1)$ | 10.1838 | $[13,1]$ | $(5,7)$ | 9.0686 | $[13,1]$ | $(7,10)$ | 8.6039 |
| | | 0.075 | $[9,3]$ | $(1,1)$ | 6.9509 | $[10,4]$ | $(9,10)$ | 6.5099 | $[11,3]$ | $(12,14)$ | 6.2959 |
| | | 0.10 | $[7,3]$ | $(1,1)$ | 5.2185 | $[8,5]$ | $(9,11)$ | 4.9492 | $[9,5]$ | $(13,14)$ | 4.8775 |
| | 2 | 0.05 | $[13,1]$ | $(1,1)$ | 10.1838 | $[13,1]$ | $(7,10)$ | 8.5707 | $[13,1]$ | $(12,14)$ | 7.9388 |
| | | 0.075 | $[9,3]$ | $(1,1)$ | 6.9509 | $[11,3]$ | $(12,14)$ | 6.2804 | $[11,3]$ | $(13,14)$ | 5.8745 |
| | | 0.10 | $[7,3]$ | $(1,1)$ | 5.2185 | $[9,5]$ | $(13,14)$ | 4.8714 | $[9,5]$ | $(13,14)$ | 4.5977 |
| $\underline{r} = (0.25, 0.125)$ | | | | | | | | | | | |
| 1 | 1 | 0.05 | $[8,3]$ | $(1,1)$ | 6.6525 | $[8,4]$ | $(5,5)$ | 6.0695 | $[8,5]$ | $(7,7)$ | 5.8679 |
| | | 0.075 | $[6,2]$ | $(1,1)$ | 4.4527 | $[5,4]$ | $(4,4)$ | 3.8974 | $[5,4]$ | $(5,5)$ | 3.6552 |
| | | 0.10 | $[4,2]$ | $(1,1)$ | 3.1785 | $[4,2]$ | $(1,3)$ | 2.7359 | $[4,2]$ | $(4,4)$ | 2.5211 |
| | 2 | 0.05 | $[8,3]$ | $(1,1)$ | 6.6525 | $[8,5]$ | $(7,7)$ | 5.8587 | $[8,5]$ | $(9,9)$ | 5.4209 |
| | | 0.075 | $[6,2]$ | $(1,1)$ | 4.4527 | $[5,3]$ | $(1,5)$ | 3.5254 | $[5,4]$ | $(6,7)$ | 3.3174 |
| | | 0.10 | $[4,2]$ | $(1,1)$ | 3.1785 | $[4,2]$ | $(1,4)$ | 2.5387 | $[4,2]$ | $(5,6)$ | 2.2780 |
| 0.5 | 1 | 0.05 | $[12,2]$ | $(1,1)$ | 10.1016 | $[11,3]$ | $(4,6)$ | 8.8776 | $[11,3]$ | $(6,8)$ | 8.4141 |
| | | 0.075 | $[8,3]$ | $(1,1)$ | 6.7758 | $[8,4]$ | $(6,6)$ | 6.1325 | $[8,5]$ | $(8,9)$ | 5.9258 |
| | | 0.10 | $[6,2]$ | $(1,1)$ | 4.9274 | $[6,3]$ | $(4,5)$ | 4.4154 | $[7,3]$ | $(7,10)$ | 4.3951 |
| | 2 | 0.05 | $[12,2]$ | $(1,1)$ | 10.1016 | $[11,3]$ | $(5,9)$ | 8.3872 | $[11,3]$ | $(10,14)$ | 7.7659 |
| | | 0.075 | $[8,3]$ | $(1,1)$ | 6.7758 | $[8,5]$ | $(7,9)$ | 5.9073 | $[9,4]$ | $(12,13)$ | 5.5994 |
| | | 0.10 | $[6,2]$ | $(1,1)$ | 4.9274 | $[6,4]$ | $(5,8)$ | 4.2553 | $[7,3]$ | $(9,10)$ | 4.0371 |

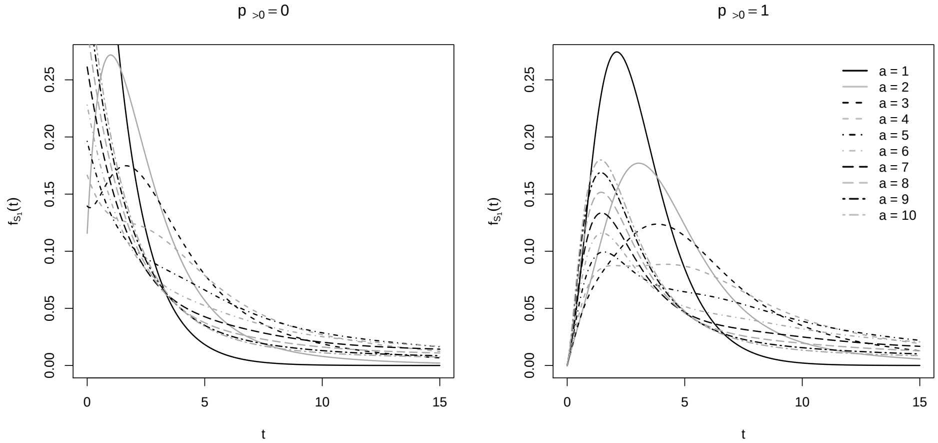

Figure 3.8 (a)

Figure 3.8 (b)

Figure 3.8: Plots of class-1 sojourn time densities for $(a, a)$ threshold policies, $a = 1, 2, \ldots, 10$, with Exp service, $M_{\mathrm{B}} = M_{\mathrm{S}} = 1$, $p_{>0} = 0, 1$, $C = 8$, $f = 2$, and $\alpha = 0.05$.

We still elect to use the switch-in time distributions defined in Equations (3.14)-(3.17), as the size of the state space is less sensitive to the number of switching phases, as they are not always tracked like service phases are by $Y_1$ and $Y_2$.

In Figure 3.8, we plot $f_{\mathcal{S}_1}(t)$ for $t \in [0, 15]$ and $a = 1, 2, \ldots, 10$, letting $\alpha = 0.05$ and $M_{\mathrm{B}} = M_{\mathrm{S}} = 1$. We consider both $p_{>0} = 0$ and $p_{>0} = 1$ to visualize the impact of switch-in times. Upon first inspection, it is clearly evident that the densities differ greatly for low values of $a$, when class 1's relative priority to class 2 is at its highest, while its shape is more consistent at higher values of $a$, requiring larger queue lengths (which are rarer to observe) and hence reducing the threshold's impact. Unsurprisingly, as we are considering class-1 sojourn times, the lower threshold policies result in more density towards small sojourn times and have lighter tails. Letting $p_{>0} = 0$, sojourn times for a class-1 machine will be shortened on average due to not having to potentially wait for the server to switch depending on the state of the system at the failure epoch, as well as having fewer class-1 machines queued ahead of it caused by the system's higher rate at which machines are repaired, $\lambda_r^{[C,f]}$, as a consequence of the server never being idle when there are still broken machines to repair (as observed in Table 3.2 for a range of policies).

Of these plots, the $(4, 4)$ threshold policy stands out as having a particularly interesting density, exhibiting a bimodal structure in the $p_{>0} = 1$ case as it has two local maxima. A sojourn time of a machine will depend greatly on the initial state of the system immediately after its failure epoch, particularly on the location of the server, so we decompose the density $f_{\mathcal{S}_1}(t)$ into components $f_{\mathcal{S}_1, L_I}(t)$ where $L_I \in \{1, 2, 3, 4\}$ are the possible server locations after observing the failure. We achieve this decomposition by considering each case separately, modifying Equation (3.8) (and hence $\underline{\Phi}_1$) by setting any element $p_{m,n,l,y,y_1,y_2}$ of the probability vector with $l \neq L_I$ equal to zero. If we re-normalized the modified $\underline{\Phi}_1$'s, then this would alternatively result in the conditional distributions of a class-1 sojourn time given different initial server locations.

Due to the nature of our considered preemptive resume threshold policies, sojourn times when $L_I = 3$ (class-2 switch-in) or $L_I = 4$ (class-2 service) will be comparable in the majority of cases due to a class-2 switch-in time being small relative to a service time and the threshold

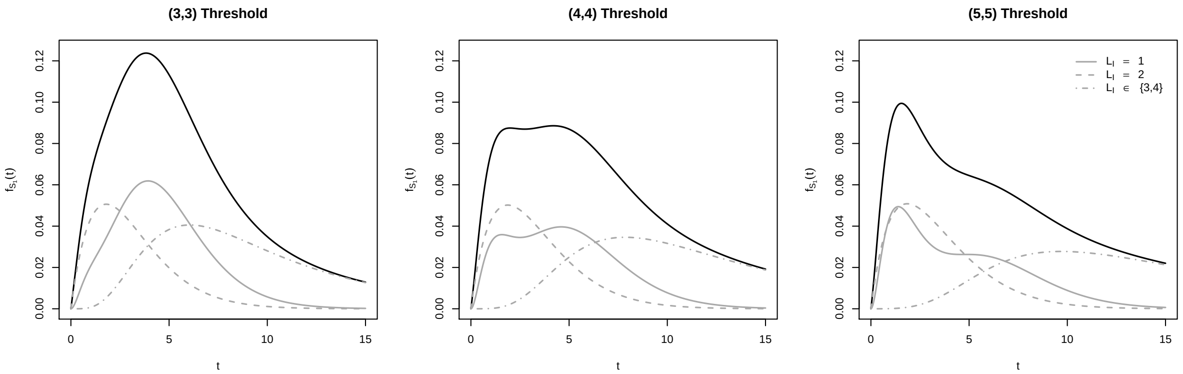

Figure 3.9 (a)

Figure 3.9 (b)

Figure 3.9 (c)

Figure 3.9: Plots of class-1 sojourn time densities and their component densities $f_{\mathcal{S}_1,L_I}$ for $(a,a)$ threshold policies, $a = 3,4,5$, with Exp service, $M_{\text{B}} = M_{\text{S}} = 1$, $p_{>0} = 1$, $C = 8$, $f = 2$, and $\alpha = 0.05$.

being commonly triggered prior to the next class-2 service completion. Thus, we keep these two cases grouped together, leaving us with $L_I = 1$, $L_I = 2$, and $L_I \in \{3,4\}$, so that

$$f_{\mathcal{S}_1}(t) = f_{\mathcal{S}_1,1}(t) + f_{\mathcal{S}_1,2}(t) + f_{\mathcal{S}_1,\{3,4\}}(t),\ t > 0.$$

In Figure 3.9, we plot the densities and their three components for $a = 3,4,5$. We observe that the components $f_{\mathcal{S}_1,2}(t)$ are very comparable, whereas $f_{\mathcal{S}_1,\{3,4\}}(t)$ has its density allocated to larger sojourn times as $a$ increases, representing the requirement of more total class-1 machine failures to trigger the higher thresholds (which also increases the probability of needing to wait for one or more class-2 repairs to complete prior to receiving service). Note that $f_{\mathcal{S}_1,\{3,4\}}(t)$ appears to be solely responsible for the remaining tails of these distributions, as the machine will almost surely be repaired within 15 time units if the server is either already serving class 1 or is switching to class 1 after the target machine fails.

It is in $f_{\mathcal{S}_1,1}(t)$ that we observe great variability between the adjacent thresholds, including the bimodal structure observed in the $(4,4)$ threshold policy. We note that this second local maxima is near 5 time units. If the target machine triggered the threshold, then it would take on average 2 time units for the class-1 switch-in time and 4 time units to repair the target machine and the three machines queued ahead of it, for a total of 6 time units. In fact, plotting the density of the sojourn time in this specific case (which we omit here) results in a right-skewed density possessing a single maxima just after $t = 5$. We therefore suspect a large portion of initial states to be of this type, causing the observed second maxima.

Letting $p_{m,\bullet,1,\bullet,\bullet,\bullet} = \sum_{n,y,y_1,y_2} p_{m,n,1,y,y_1,y_2}$ denote the marginal probability of the server conducting a class-1 switch-in immediately after a class-1 machine failure fills the $m^{\text{th}}$ slot in queue 1, we compare these probabilities for $m = 1,2,\ldots,8$ and $a = 3,4,5$ in Table 3.6. It would seem that in the $L_I = 1$ cases, there is indeed a large jump in initial probability for cases where the target machine is indeed the threshold trigger. The other most likely cases denote a failure to an empty system (i.e., $m = 1$) which is more likely the higher the threshold (as it lets the class 2 queue empty faster) and at $m = a + 1$ indicating cases where the target machine failed during a switch-in time in progress which was triggered by the preceding class-1 machine failure. Other choices of $m$ have much less probability as they require there to be multiple failures during the short switch-in time.

Table 3.6: Marginal $L_I = 1$ class-1 sojourn time distribution initial probabilities, $p_{m,\bullet,1,\bullet,\bullet,\bullet}$, for the $(a, a)$ threshold service policy with $a = 3, 4, 5$, Exp service, $M_{\mathrm{B}} = M_{\mathrm{S}} = 1$, $p_{>0} = 1$, $C = 8$, $f = 2$, and $\alpha = 0.05$.

|  | $m$ |  |  |  |  |  |  |  |
|---|---|---|---|---|---|---|---|---|
| $(a, b)$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| $(3, 3)$ | 0.0395 | 0.0127 | 0.2181 | 0.0597 | 0.0106 | 0.0014 | 0.0001 | $< 0.0001$ |
| $(4, 4)$ | 0.0688 | 0.0213 | 0.0074 | 0.1512 | 0.0376 | 0.0058 | 0.0006 | $< 0.0001$ |
| $(5, 5)$ | 0.0992 | 0.0300 | 0.0095 | 0.0049 | 0.1076 | 0.0228 | 0.0028 | 0.0002 |

We therefore conclude that the jumps in $p_{m,\bullet,1,\bullet,\bullet,\bullet}$ near $m = 1$ and $m = a$ are responsible for the shapes of density components $f_{\mathcal{S}_1,1}(t)$ (as they of course represent mixtures of distributions), namely the bimodal structure of the $(4, 4)$ threshold policy as well as the flat region in the $(5, 5)$ threshold policy. For the $(3, 3)$ threshold policy, $p_{3,\bullet,1,\bullet,\bullet,\bullet}$ is much larger than $p_{1,\bullet,1,\bullet,\bullet,\bullet}$, which hides this obvious mixture appearance.

### 3.5.3 Smart Bernoulli Optimization

Among other service policies, we considered $(1, 0.2)$ and $(1, 0.8)$ smart Bernoulli in Table 3.2 and Figures 3.4 and 3.5. It was evident that due to $(1, 0.2)$ smart Bernoulli's higher preference for serving class-1 machines (causing additional switch-ins), it both converged to $\mathrm{E}[N_{\mathrm{W}}^{[\infty]}]$ at fewer total machines when switch-in times were identically zero in duration, and to a lower limit when switch-in time durations had positive expected values, relative to $(1, 0.8)$ smart Bernoulli. In Table 3.2, $(1, 0.2)$ had a larger $\mathrm{E}[N_{\mathrm{W}}^{[C,f]}]$ in every considered case except when $p_{>0} = 1$ and $[C, f] = [14, 0]$. In this subsection, we will investigate some new examples to observe the impact of switch-ins and the number of machines on the optimal selection of smart Bernoulli probabilities that maximizes $\mathrm{E}[N_{\mathrm{W}}^{[C,f]}]$.

First of all, we justify the choice of $p_1^{\mathrm{SB}} = 1$. The $c\mu$ rule (Meilijson and Yechiali [68], van Mieghem [91]) states that in a priority queue, if class-$i$ customers have a holding cost of $c_i$ per time unit and an expected service time of $1/\mu_i$, then the classes should be served in decreasing order of $c_i\mu_i$, independent of arrival rate. For finite-population systems, this is not necessarily true, as the presence of a broken machine waiting to be serviced reduces the number of machines that can fail of that type (i.e., despite a potentially fast service time, if each class of machines comes from its own independent population and the time to failure for class-$i$ machines is small, then it may not be optimal to give them higher service priority). A modified $c\mu\lambda$ rule (Iravani and Kolfal [47]) was investigated for a fully exponential model of this type (an example of the machine-repairman problem). Based on certain assumptions and conditions, it was concluded that priority may be given to class $j$ with positive queue length if $\frac{c_j\mu_j}{\lambda_j} \geq \frac{c_k\mu_k}{\lambda_k}$ $\forall\ k \neq j$, such that there is at least one class-$k$ machine waiting to be repaired.

In our model, since both classes of failure come from the same pool of machines, we can effectively ignore the fact that we are using a finite-population system since no matter which machine type is repaired, the time until it fails again has an identical distribution. Thus, we can consider the standard $c\mu$ rule. For our model under the case of zero duration switch-in times, we would assign priority to the class with the highest value of $c_i\mu_i$, and as such always prefer to serve it over the other class. In our investigation, we simply want to maximize the

Table 3.7: Expected values and variances for service time distributions Exp, $H_2$, and $E_3$, along with the LN distributions of interest and their EM algorithm fits using continuous phase-type distributions of order 5.

| | Class 1 | | Class 2 | |
|---|---|---|---|---|
| Service | Expectation | Variance | Expectation | Variance |
| $H_2$ | 1 | 5.5 | 20 | 2200 |
| Exp | 1 | 1 | 20 | 400 |
| $E_3$ | 1 | 1/3 | 20 | 400/3 |
| LN | 1 | 4 | 20 | 2000 |
| LN (fit) | 0.99998 | 3.80365 | 19.97997 | 1227.85435 |

expected number of working machines, so we would select equal holding costs (e.g., $c_1 = c_2 = 1$), as a broken machine of either type equally lowers the expected number of working machines. Therefore, by the $c\mu$ rule, the class with the highest $\mu_i$ (i.e., shortest expected service time) should have priority, corresponding to class 1 in our numerical examples.

Now, if in this zero switch-in case we would never want to switch away from class 1 (to go serve class 2), then in the cases with positive switch-in times, it follows that it would still never be optimal to switch away from class 1 since not only would the mechanic switch to serving the less efficient-to-serve class, they must incur a period of idleness during the switch-in which reduces their average rate of repair. Thus, similar to the arguments of Blanc and van der Mei [14], we can conclude that in the smart Bernoulli framework, class 1 (having the smallest average repair times) should receive a probability of $p_1^{\mathrm{SB}} = 1$ to continue repairs (and hence, not switching) after each service completion, should its queue not be empty.

It is not as clear for the lower priority class 2. If there were no switch-in times, then it would be optimal to switch after every service completion and have a probability of starting another service of $p_2^{\mathrm{SB}} = 0$. However, as each positive duration switch-in time incurs idleness, in reality there may be an optimal $p_2^{\mathrm{SB}}$ that is positive. This probability is what we must find to optimize the use of smart Bernoulli in our model. To do this, we find the approximate $\hat{p}_2^{\mathrm{SB}}$ that maximizes $\mathrm{E}[N_{\mathrm{W}}]$ using the algorithm outlined in the Appendix. For all approximated optimal $\hat{p}_2^{\mathrm{SB}}$ in this subsection, we set precision = 4 (i.e., we approximate to four decimal places).

We now investigate the impacts of reducing $p_1^{\mathrm{SB}}$ from 1 (considering $p_1^{\mathrm{SB}} \in \{0.9, 0.95, 1\}$) and varying the expected switch-in time durations in Figure 3.10, where we plot the optimal $p_2^{\mathrm{SB}}$ against $p_{>0}$ (with $M_{\mathrm{S}} = 1$) or $M_{\mathrm{S}}$ (with $p_{>0} = 1$), so that the mean switch-in time durations are equal and hence comparable. The corresponding values of $\mathrm{E}[N_{\mathrm{W}}]$ calculated using the optimal values of $p_2^{\mathrm{SB}}$ for the $\alpha = 0.10$ cases are plotted in Figure 3.11. We set $M_{\mathrm{B}} = 1$ and allow both class' service time distributions to be Exp, $H_2$, or $E_3$, to observe the effect of service variance, while letting $\alpha \in \{0.075, 0.10\}$, $C = 8$, and $f = 2$. Additionally, we approximate the impact of heavy-tailed service time distributions by applying the EM algorithm (Asmussen et al. [7]) to fit log-normal (LN) distributions to continuous phase-type distributions of order 5. A summary of the service time distributions' expectations and variances are provided in Table 3.7. Log-normal parameters were selected to match mean repair time values, while being slightly less variable than the $H_2$ distributions. While the approximations of these LN distributions provide very close fits for the expected values, the difficulty of accurately fitting heavy tails is evident by their smaller variances.

In Figure 3.10, we can see that for very small switch-in times it is optimal to maintain $p_2^{\mathrm{SB}} = 0$ and act as a class-1 non-preemptive priority policy (or similar to one if $p_1^{\mathrm{SB}} < 1$), but by increasing the mean switch-in times we make additional switches (relative to the exhaustive service policy) more costly and it becomes optimal for $p_2^{\mathrm{SB}}$ to become positive, eventually reaching $p_2^{\mathrm{SB}} = 1$ in order to minimize the number of switch-ins (note that in the $p_1^{\mathrm{SB}} = 1$ and $\alpha = 0.075$ case, if we continue to increase $M_{\mathrm{S}}$ beyond 1, then these curves will also hit $p_2^{\mathrm{SB}} = 1$).

By decreasing $p_1^{\mathrm{SB}}$, it becomes possible to switch away from class 1 before its queue empties and it is not hard to see that this will have the effect of increasing the fraction of time that the mechanic is idle. This has the effect of increasing the slopes of the curves in Figure 3.10, indicating that the behaviour dictating how the mechanic treats class 2 is more sensitive to the expected switch-in time durations and will opt to treat class 2 in an exhaustive manner sooner, even if they are not allowed to do the same for class 1, in order to compensate for the additional class-2 switch-ins out of class 1.

By decreasing $\alpha$, there are fewer failures resulting in shorter queue lengths and less opportunities for the smart Bernoulli policy to cause the server to leave before emptying a queue. Therefore, increasing $p_2^{\mathrm{SB}}$ has a smaller impact on reducing the number of extra switch-ins and the mean switch-in durations need to be larger before it becomes optimal to use a positive $p_2^{\mathrm{SB}}$.

Comparing the four sets of service time distributions, they transition from $p_2^{\mathrm{SB}} = 0$ to $p_2^{\mathrm{SB}} = 1$ at comparable rates, but the more variable distributions require more incentive in the form of higher costs from switch-in times to increase $p_2^{\mathrm{SB}}$ from 0. This follows since the more class-2 services that are completed before returning to the class-1 queue, the more opportunities there are for the server to be stuck in a particularly long service time (e.g., the 10% case in the class-2 $\mathrm{H}_2$ distribution having mean 110) which will have a large effect on the sojourn times of class-1 machines that are waiting to be serviced. As service variance is reduced, there is less uncertainty accepted from additional class-2 service times and the mechanic is willing to begin increasing $p_2^{\mathrm{SB}}$ at smaller mean switch-in times. We observe that the $\mathrm{H}_2$ and LN service time distributions result in very close optimal values of $p_2^{\mathrm{SB}}$. As $\mathrm{H}_2$ is more variable, this would indicate that optimality must be less sensitive to changes in variance when it is already large.

In Figure 3.11, we confirm that reducing $p_1^{\mathrm{SB}}$ lowers the maximum $\mathrm{E}[N_{\mathrm{W}}]$ possible at the corresponding optimal $p_2^{\mathrm{SB}}$ probabilities. The differences between the plots may not be large, but note that these are not for fixed $p_2^{\mathrm{SB}}$, but rather the optimal $p_2^{\mathrm{SB}}$'s at each $p_{>0}$ or $M_{\mathrm{S}}$ given the different values of $p_1^{\mathrm{SB}}$. Additionally, we observe that increasing service variance has a negative effect on the mean number of working machines (e.g., $\mathrm{H}_2$ has lower values than LN, despite similar optimal $p_2^{\mathrm{SB}}$ probabilities), but the relationship between $\mathrm{E}[N_{\mathrm{W}}]$ and switch-in times is primarily dependent on the first moments. Interestingly, these relationships are approximately linear between $\mathrm{E}[N_{\mathrm{W}}]$ and the mean switch-in times in ranges where the optimal $p_2^{\mathrm{SB}}$ are unchanged, either at 0 or 1.

In Figure 3.12, we plot optimal $p_2^{\mathrm{SB}}$ against $C = 3, 4, \ldots, 25$ for $f = 0, 2, 4$, $p_1^{\mathrm{SB}} = 1$, $M_{\mathrm{B}} = M_{\mathrm{S}} = 1$, $p_{>0} = 0.5$, and $\alpha \in \{0.05, 0.075, 0.10\}$. We observe that increasing $C$ results in more failures, longer queue lengths, and more opportunities for a smart Bernoulli policy to cause a switch from a queue before it is emptied. Therefore, as $C$ becomes large, with the exception of the fitted LN distributions, the server eventually increases $p_2^{\mathrm{SB}}$ until class 2 is treated in an exhaustive manner. Increasing $f$ has a similar effect, and as the total number of machines are greater for a given $C$, the mechanic begins the transition from class-1 non-preemptive priority to an exhaustive policy at fewer $C$, acting largely as a horizontal shift with minimal effect on the rate of increase in $p_2^{\mathrm{SB}}$. By increasing $\alpha$, the sensitivity of the optimal $p_2^{\mathrm{SB}}$ on $C$ is heightened as every increment of $C$ has a larger impact on the average failure rate, causing $p_2^{\mathrm{SB}}$ to transition from 0 to 1 at fewer total machines and at a faster rate. Finally, in comparing the Exp, $\mathrm{H}_2$, and $\mathrm{E}_3$ service time distributions, we observe results consistent with those from Figure 3.10, in that it becomes optimal to increase $p_2^{\mathrm{SB}}$ earlier (i.e., for smaller $C$) for service time distributions having smaller variances.

While the fitted LN distributions acted very similarly to the $\mathrm{H}_2$ distributions in Figure 3.10, they display a unique behaviour in Figure 3.12. In Figure 3.12, rather than converging to an exhaustive discipline as $C$ increases, the optimal $p_2^{\mathrm{SB}}$ peaks before decreasing to some positive limit. This peak seems highest for the cases with $f = 0$, falling just short of 1 in Figure 3.12 (d). In part (f), all three plots hit $p_2^{\mathrm{SB}} = 1$ before moving away from the exhaustive policy at 20 total machines. As this behaviour is not shared with the other pair of highly variable service time distributions, this seems to suggest that it must be due to the more general structure of the fitted continuous phase-type distributions. As these are intended to behave similarly to heavy-tailed distributions, it would be of great interest to revisit this problem using a like model within a semi-Markov framework. This would allow the usage of general distributions for service and switch-in times, and hence, enable us to investigate the true impact of heavy-tailed distributions.

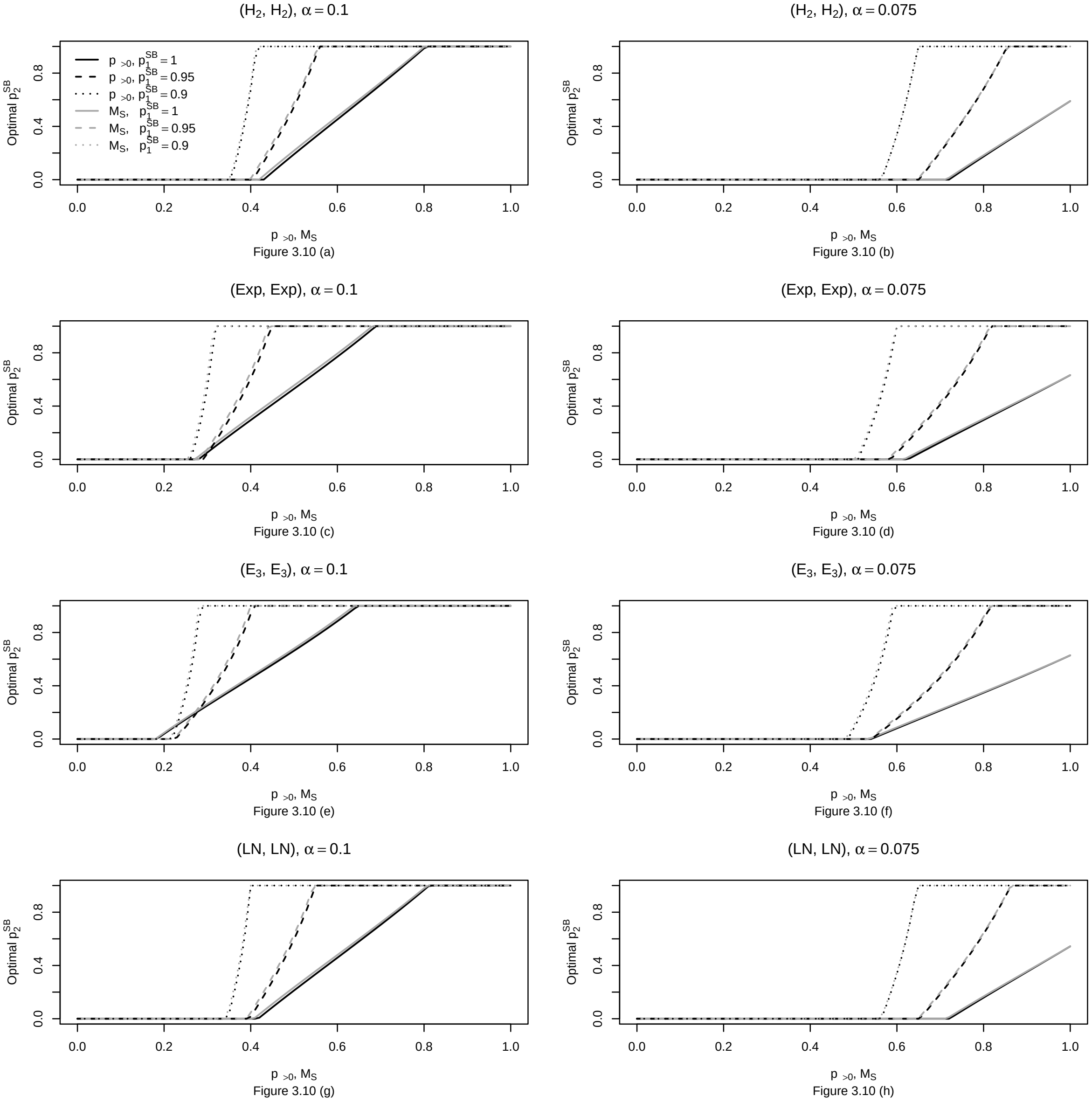

Figure 3.10 (a)

Figure 3.10 (b)

Figure 3.10 (c)

Figure 3.10 (d)

Figure 3.10 (e)

Figure 3.10 (f)

Figure 3.10 (g)

Figure 3.10 (h)

Figure 3.10: Plots of optimal class-2 smart Bernoulli probability $p_2^{\mathrm{SB}}$ against $p_{>0}$ (with $M_{\mathrm{S}} = 1$) or $M_{\mathrm{S}}$ (with $p_{>0} = 1$) for $\alpha = 0.075, 0.1$ and $p_1^{\mathrm{SB}} = 0.9, 0.95, 1$.

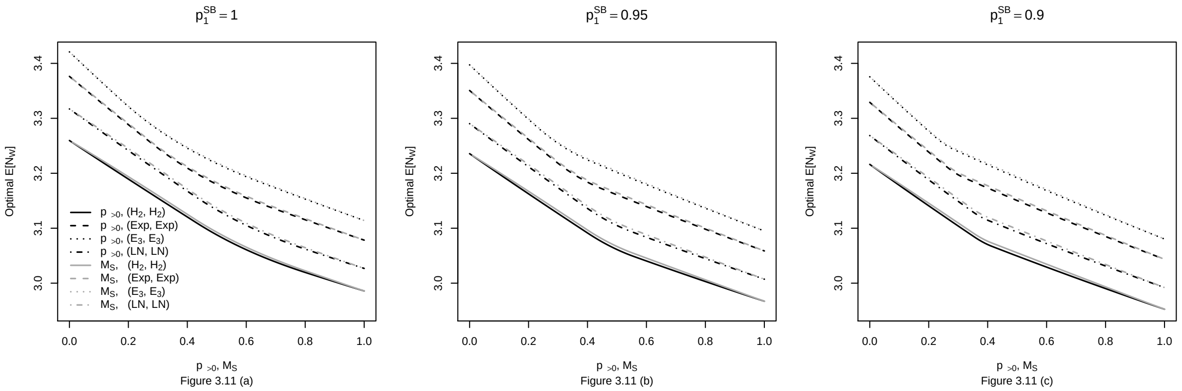

Figure 3.11: Plots of $\mathrm{E}[N_\mathrm{W}]$ at optimal class-2 smart Bernoulli probabilities against $p_{>0}$ (with $M_\mathrm{S} = 1$) or $M_\mathrm{S}$ (with $p_{>0} = 1$) for $\alpha = 0.10$ and $p_1^\mathrm{SB} = 0.9, 0.95, 1$.

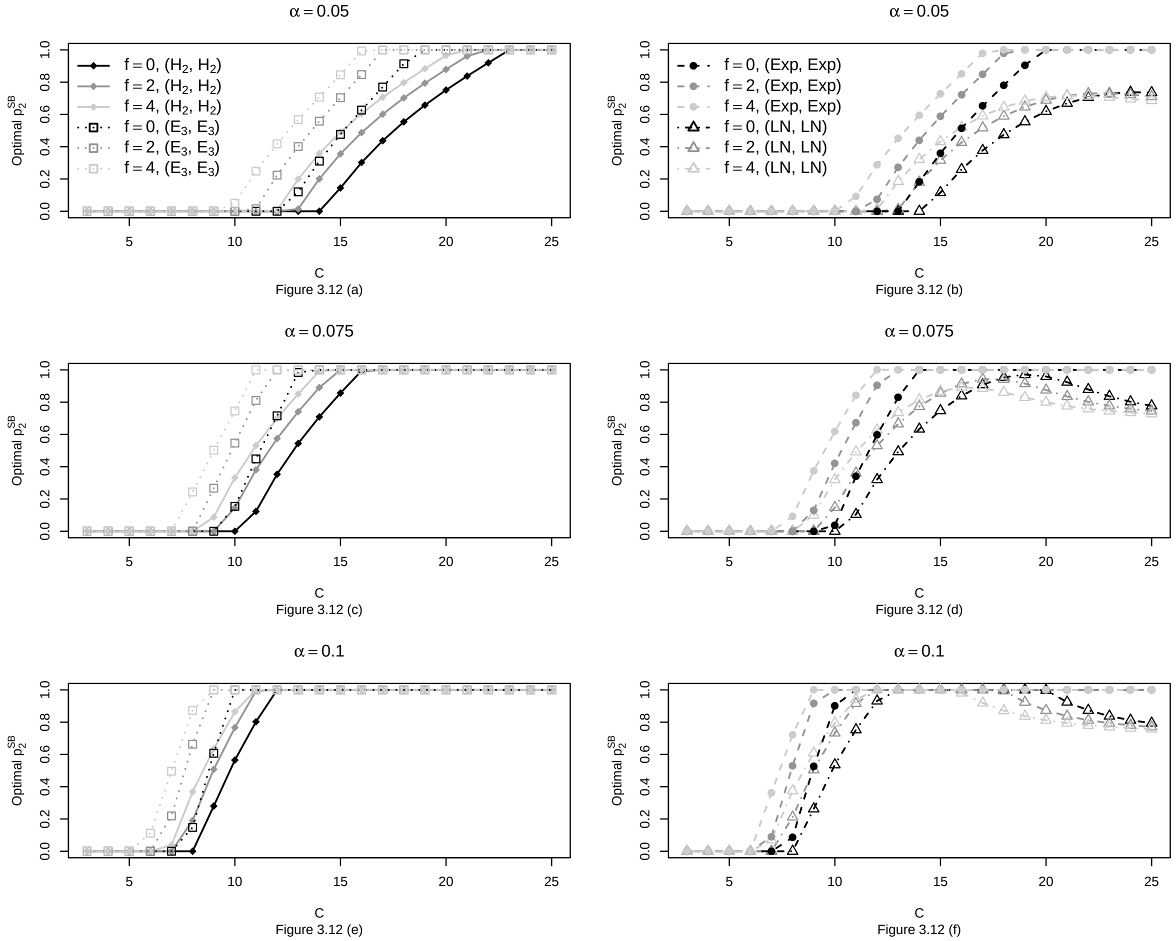

Figure 3.12: Plots of optimal class-2 smart Bernoulli probability $p_2^\mathrm{SB}$ against $C$ for $f = 0, 2, 4$, $\alpha = 0.05, 0.075, 0.10$, $p_{>0} = 0.5$, $p_1^\mathrm{SB} = 1$, and $M_\mathrm{B} = M_\mathrm{S} = 1$.

# Part II : Infinite Population Cyclic Polling Models

# Chapter 4

# A 2-Class Polling Model with Class-Dependent Reneging, Switchover Times, and Phase-Type Service

## 4.1 Discussion of Literature

In Chapters 2 and 3 of this thesis, we considered a range of service disciplines including exhaustive, preemptive and non-preemptive priority, threshold, and (smart) Bernoulli. Within this chapter, we specifically consider an application of the $k_i$-limited discipline introduced in Section 1.2.7, where we demonstrated some related structures in Example 7. Similar to threshold and Bernoulli disciplines, a $k_i$-limited service discipline allows us to assign a form of relative priority to a queue. This makes it attractive from the vantage of system optimization.

Unfortunately, as mentioned in Section 1.2.8, the $k_i$-limited discipline does not satisfy the branching property. This has the result of making it much more difficult, if not impossible, to analyze exactly in complicated models. When exact analysis of a polling model with $k_i$-limited service is conducted, it is typically in the case of a 2-queue system. Even then, restrictions are required. Chang and Down [28] considered a 2-queue polling system with $k_i$-limited service disciplines at both queues. However, they required the restrictive assumption that the service times for both classes are exponentially distributed with the same rate. They did allow the Poisson process arrival rates and selected $k_i$ values to differ. The derived exact asymptotic probabilities for the event of there being $l$ total customers in the system. Two cases were considered, where the event of the system holding a large number of customers is primarily caused by a single class, or both classes together.

A more common case permitting exact analysis assumes that service at one queue is exhaustive, while the other has the $k_i$-limited discipline. This restriction is in fact not unreasonable if the 2-queue system allows the exhaustive queue to represent higher priority customers or jobs, as this would agree with Boxma et al. [17] in terms of optimal server resources. Similarly, this is in agreement with what we observed for our smart Bernoulli optimization in Section 3.5.3.

The following three papers all allowed for a 2-queue system of this type, with Poisson process arrivals and general service. Ozawa [72] solved for the mean waiting times of customers by using a piecewise Markov process to model the number of customers in the $k_i$-limited queue, resulting in the derivation of the PGF of the number of class-2 customers in their queue at arbitrary times. The waiting time of the exhaustive queue was found from the fact that the model was work conserving, and the mean waiting times for the other class of customers was known. Lee [59] went a bit further, deriving the PGFs of the number of customers at departure instants for both classes. From these PGFs, the LSTs of customer sojourn times for both classes were obtained. Winands et al. [97] solved for the marginal PGFs of queue lengths at steady state for both queues in the same kind of model, however they assumed the presence of generally distributed setup times which were absent from these other two models. Interestingly, they allowed the setup times to only be conducted if that queue at the polling instant had a positive length, which is a more realistic assumption for a real world production system.

When considering a polling system that uses this discipline while having a general number of queues, approximations may be substituted in place of exact analysis. Borst et al. [17], as previously discussed in Section 1.2.8, derived four approximations for how to calculate mean waiting times in a system with $k_i$-limited service, a general number of queues, Poisson process arrivals, and generally distributed service and switchover times. Such a model was also considered by van Vuuren and Winands [92], who had an interest in the marginal queue length distributions. They made use of an iterative approximation which decomposed the queue into single queue systems with vacations (a similar idea as Servi [85], who considered Bernoulli disciplines). In this way, any time spent away from a class's queue was treated as a generally distributed vacation from the perspective of that class. In order to improve the accuracy of their approximations, the distributions of vacations from each class were allowed to depend on the number of services (up to $k_i$) that were completed during the most recent visit, and they showed that considerable relative accuracy gains were obtained by taking such correlations into account.

If we forgo the possibility of generally distributed service and switchover times, numerical methods can be used to obtain very accurate approximations. For example, Blanc [13] demonstrated how to apply their power-series algorithm to calculate the steady-state distribution of a cyclic polling model having a general number of queues and $k_i$-limited service under the assumption of Poisson process arrivals and exponentially distributed service times. Of course, another option for such an analysis is MAM, which we will now demonstrate in action for a 2-queue system in this chapter, and in a $N$-queue system in Chapter 6. We close this subsection by remarking that a majority of the work within this chapter may be found in Granville and Drekic [39].

## 4.2 Model Assumptions

We consider a polling model in which a single server provides service to two distinct classes of customers, each having its own respective queue. Customers are served on a FCFS basis within their own queue. Let $C_i < \infty$ be the class-$i$ buffer size, $i = 1, 2$. Customers of classes 1 and 2 arrive to the system according to independent Poisson processes with rates $\lambda_1$ and $\lambda_2$, respectively. Service times for class-$i$ customers, $i = 1, 2$, are assumed to have a continuous phase-type distribution with representation $Ser_i \sim \text{PH}_{b_i}(\underline{\beta}_i, B_i)$, and we assume that a customer's service time is independent of all other service times as well as the arrival processes. For $i = 1, 2$, let $\mu_i$ denote the mean class-$i$ service time.

Service is administered according to the $k_i$-limited service discipline, in which the server

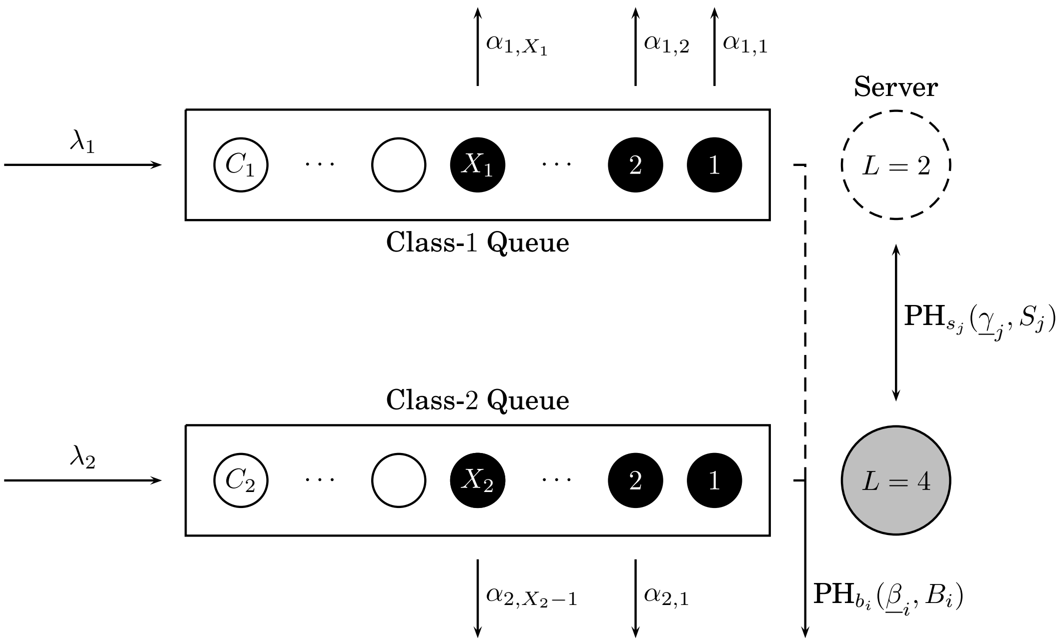

Figure 4.1: Depiction of the polling model during a sojourn of the server at queue 2.

serves up to $k_i$ customers of class $i$, switching over to the other class once the class-$i$ queue empties or the maximum number of services has been reached. Note that by letting $k_i \to \infty$ for $i = 1, 2$, it is possible to model a 2-class polling model with *exhaustive* service. Moreover, we can capture the class-1 (class-2) non-preemptive priority service discipline by letting $k_1 \to \infty$ and $k_2 = 1$ ($k_1 = 1$ and $k_2 \to \infty$). Once the decision to switch out of class $i$ has been made, the server initiates a class-$j$ switch-in time, $j \neq i$, which has a continuous phase-type distribution with representation $\text{PH}_{s_j}(\underline{\gamma}_j, S_j)$, which we assume to be strictly positive in duration. Switch-in times are independent of each other, as well as service times and the arrival processes. Furthermore, we assume that the server is unable to determine whether the other class is empty before initiating a switch, so it is possible for multiple switches to take place before the server finally encounters a customer waiting to be served. As a result, the server is never truly idle in the system, even when both queues are empty.

We also incorporate the notion of class-dependent reneging and assume that when an arriving class-$i$ customer enters the system, it leaves the system following an (independent) exponentially distributed amount of time with a rate that can change along with their position in their queue, $\alpha_{i,n}$. A customer who reneges from the system is considered lost. Here, the '$n$' in $\alpha_{i,n}$ indicates that they have $n-1$ other class-$i$ customers waiting for service in front of them. Once a customer does reach the server, however, we assume that customer is no longer subject to reneging. For notational convenience, we define $\alpha_i^{[j]} = \sum_{n=1}^{j} \alpha_{i,n}$ to be the total force of reneging for $j$ waiting class-$i$ customers (and use the convention $\alpha_i^{[0]} = 0$).

A graphical illustration of the polling model during a visit by the server to the class-2 queue is given in Figure 4.1. Customers present in the system are represented by solid black circles and empty circles with solid black outlines represent open slots in either queue available to future arriving customers, while the solid grey circle and dashed empty circle represent the current and potential locations that the server can work, with locations denoted by $L$ which will be defined in Section 4.3. Note that unlike the previously considered maintenance models, there is no location for the server to idle in this polling model. In this example, the leading class-2 customer will depart after a random $\text{PH}_{b_2}(\underline{\beta}_2, B_2)$ amount of time, while every other customer who is waiting has an active impatience timer whose exponential rate depends on their position in their respective queue.

## 4.3 Determination of the Steady-State Probabilities

We model this polling model using CTMC

$$\{(X_1(t), X_2(t), L(t), K(t), Y(t)), t \geq 0\},$$

where $X_i(t)$ represents the number of class-$i$ customers present in the system, $i = 1, 2$, such that $X_i(t) \in \{0, 1, \ldots, C_i\}$. We again let $L(t) \in \{1, 2, 3, 4\}$ denote the location of the server, where $L(t) = 2i - 1$ represents switching into class $i$ and $L(t) = 2i$ represents serving class $i$, $i = 1, 2$. The possible values of $L$ that the CTMC can take depends on both queue lengths, such that

$$L(t) \in \Omega_L(X_1(t), X_2(t)) = \begin{cases} \{1,3\} & , \text{ if } X_1(t) = 0,\ X_2(t) = 0, \\ \{1,2,3\} & , \text{ if } X_1(t) > 0,\ X_2(t) = 0, \\ \{1,3,4\} & , \text{ if } X_1(t) = 0,\ X_2(t) > 0, \\ \{1,2,3,4\} & , \text{ if } X_1(t) > 0,\ X_2(t) > 0. \end{cases} \tag{4.1}$$

To enable our $k_i$-limited service discipline, $K(t)$ represents the server being on their $k^{\text{th}}$ service within a visit to a queue (where we let $K = 0$ if the server is undergoing a switch-in time), where

$$K(t) \in \Omega_K(L(t)) = \begin{cases} \{0\} & , \text{ if } L(t) = 1, \\ \{1,2,\ldots,k_1\} & , \text{ if } L(t) = 2, \\ \{0\} & , \text{ if } L(t) = 3, \\ \{1,2,\ldots,k_2\} & , \text{ if } L(t) = 4. \end{cases} \tag{4.2}$$

Lastly, $Y(t)$ denotes the current phase of a service or switch-in time, similarly depending on $L(t)$ such that

$$Y(t) \in \Omega_Y(L(t)) = \begin{cases} \{1,2,\ldots,s_1\} & , \text{ if } L(t) = 1, \\ \{1,2,\ldots,b_1\} & , \text{ if } L(t) = 2, \\ \{1,2,\ldots,s_2\} & , \text{ if } L(t) = 3, \\ \{1,2,\ldots,b_2\} & , \text{ if } L(t) = 4. \end{cases} \tag{4.3}$$

Our first objective is to determine $P_{m,n}$, the steady-state joint probability that $X_1(t) = m$ and $X_2(t) = n$ for $m = 0, 1, \ldots, C_1$ and $n = 0, 1, \ldots, C_2$. Let $\pi_{m,n,l,k,y}$ represent the steady-state joint probability of observing the CTMC in state $(m, n, l, k, y)$. Note that by Equation (4.1), when $X_1(t) = X_2(t) = 0$ (i.e., both queues are empty), it is only possible to observe the server conducting a switch from one queue to the other (as there are no customers to server in either queue), and so

$$P_{0,0} = \sum_{i=1}^{2} \sum_{y=1}^{s_i} \pi_{0,0,2i-1,0,y}.$$

Furthermore, it is an immediate consequence that

$$P_{0,n} = \sum_{i=1}^{2} \sum_{y=1}^{s_i} \pi_{0,n,2i-1,0,y} + \sum_{k=1}^{k_2} \sum_{y=1}^{b_2} \pi_{0,n,4,k,y}, \; n \geq 1,$$

$$P_{m,0} = \sum_{i=1}^{2} \sum_{y=1}^{s_i} \pi_{m,0,2i-1,0,y} + \sum_{k=1}^{k_1} \sum_{y=1}^{b_1} \pi_{m,0,2,k,y}, \; m \geq 1,$$

and

$$P_{m,n} = \sum_{i=1}^{2} \left( \sum_{y=1}^{s_i} \pi_{m,n,2i-1,0,y} + \sum_{k=1}^{k_i} \sum_{y=1}^{b_i} \pi_{m,n,2i,k,y} \right), \; m, n \geq 1.$$

With $X_1(t)$ as the level of the process, we define the 0th steady-state probability row vector to be

$$\underline{\pi}_0 = (\underline{\pi}_{0,0}, \underline{\pi}_{0,1}, \ldots, \underline{\pi}_{0,C_2}),$$

where

$$\underline{\pi}_{0,0} = (\pi_{0,0,1,0,1}, \ldots, \pi_{0,0,1,0,s_1}, \pi_{0,0,3,0,1}, \ldots, \pi_{0,0,3,0,s_2})$$

is a row vector of size $s = s_1 + s_2$ and

$$\begin{aligned} \underline{\pi}_{0,n} = (&\pi_{0,n,1,0,1}, \ldots, \pi_{0,n,1,0,s_1}, \pi_{0,n,3,0,1}, \ldots, \pi_{0,n,3,0,s_2}, \\ &\pi_{0,n,4,1,1}, \ldots, \pi_{0,n,4,1,b_2}, \pi_{0,n,4,2,1}, \ldots, \pi_{0,n,4,k_2,b_2}) \end{aligned}$$

is a row vector of size $z_1 = s + k_2 b_2$ for $n = 1, 2, \ldots, C_2$. For $m = 1, 2, \ldots, C_1$, the $m$th steady-state probability row vector is defined as

$$\underline{\pi}_m = (\underline{\pi}_{m,0}, \underline{\pi}_{m,1}, \ldots, \underline{\pi}_{m,C_2}),$$

where

$$\begin{aligned} \underline{\pi}_{m,0} = (&\pi_{m,0,1,0,1}, \ldots, \pi_{m,0,1,0,s_1}, \pi_{m,0,2,1,1}, \ldots, \pi_{m,0,2,1,b_1}, \\ &\pi_{m,0,2,2,1}, \ldots, \pi_{m,0,2,k_1,b_1}, \pi_{m,0,3,0,1}, \ldots, \pi_{m,0,3,0,s_2}) \end{aligned}$$

is a row vector of size $s + k_1 b_1$ and

$$\begin{aligned} \underline{\pi}_{m,n} = (&\pi_{m,n,1,0,1}, \ldots, \pi_{m,n,1,0,s_1}, \pi_{m,n,2,1,1}, \ldots, \pi_{m,n,2,1,b_1}, \\ &\pi_{m,n,2,2,1}, \ldots, \pi_{m,n,2,k_1,b_1}, \pi_{m,n,3,0,1}, \ldots, \pi_{m,n,3,0,s_2}, \\ &\pi_{m,n,4,1,1}, \ldots, \pi_{m,n,4,1,b_2}, \pi_{m,n,4,2,1}, \ldots, \pi_{m,n,4,k_2,b_2}) \end{aligned}$$

is a row vector of size $z_2 = s + k_1 b_1 + k_2 b_2$ for $n = 1, 2, \ldots, C_2$. We remark that level 0 is comprised of $n_1 = s + C_2 z_1$ states, whereas each non-zero level consists of a total of $n_2 = s + k_1 b_1 + C_2 z_2$ states.

Let $\underline{\pi} = (\underline{\pi}_0, \underline{\pi}_1, \ldots, \underline{\pi}_{C_1})$ be the concatenated steady-state probability row vector having a total of $C_1+1$ levels. To determine $\underline{\pi}_m$, $m = 0, 1, \ldots, C_1$, we can apply the algorithm covered in Section 1.2.6 since as a consequence of having reneging in this model, the infinitesimal generator matrix $Q$ takes on the form of a level-dependent QBD,

$$Q = \begin{array}{c|ccccccc|} & 0 & 1 & 2 & \cdots & C_1-2 & C_1-1 & C_1 \\ 0 & Q_{0,0} & Q_{0,1} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 1 & Q_{1,0} & Q_{1,1} & Q_{1,2} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 2 & \mathbf{0} & Q_{2,1} & Q_{2,2} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ C_1-2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q_{C_1-2,C_1-2} & Q_{C_1-2,C_1-1} & \mathbf{0} \\ C_1-1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q_{C_1-1,C_1-2} & Q_{C_1-1,C_1-1} & Q_{C_1-1,C_1} \\ C_1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & Q_{C_1,C_1-1} & Q_{C_1,C_1} \end{array}, \quad (4.4)$$

where blocks $Q_{i,j}$ contain all transitions where $X_1(t)$ changes from level $i$ to level $j$ and $\mathbf{0}$ denotes an appropriately dimensioned zero matrix. The overall dimension of $Q$ is $n_1 + C_1 n_2$, as $Q_{0,0}$ is an $n_1 \times n_1$ sub-matrix, $Q_{0,1}$ is an $n_1 \times n_2$ sub-matrix, $Q_{1,0}$ is an $n_2 \times n_1$ sub-matrix, and all remaining sub-matrices are of size $n_2 \times n_2$.

We first observe that $Q_{1,2} = Q_{2,3} = \cdots = Q_{C_1-1,C_1} = \lambda_1 I_{n_2}$. In what follows, recall that $\otimes$ denotes the Kronecker product operator and $\delta_{i,j}$ is the Kronecker delta function. Also, $\underline{e}_{i,j}$ is a row vector of length $i$ with 1 as the $j^{\text{th}}$ entry and zeros everywhere else, and $\underline{e}_i$ is a row vector of $i$ ones. In addition, we again let $\underline{B}'_{0,i} = -B_i\underline{e}'$ and $\underline{S}'_{0,i} = -S_i\underline{e}'$. Finally, for further notational convenience, define $\lambda_{m,n} = (1-\delta_{m,C_1})\lambda_1 + (1-\delta_{n,C_2})\lambda_2$,

$$\zeta_{m,n,l} = \begin{cases} -\left(\lambda_{m,n} + \alpha_1^{[m]} + \alpha_2^{[n]}\right) I_{s_1} + S_1 & \text{, if } l = 1, \\ -\left(\lambda_{m,n} + \alpha_1^{[m-1]} + \alpha_2^{[n]}\right) I_{k_1 b_1} + I_{k_1} \otimes B_1 & \text{, if } l = 2, \\ -\left(\lambda_{m,n} + \alpha_1^{[m]} + \alpha_2^{[n]}\right) I_{s_2} + S_2 & \text{, if } l = 3, \\ -\left(\lambda_{m,n} + \alpha_1^{[m]} + \alpha_2^{[n-1]}\right) I_{k_2 b_2} + I_{k_2} \otimes B_2 & \text{, if } l = 4, \end{cases}$$

and

$$U_i = \begin{cases} \mathbf{0} & \text{, if } k_i = 1, \\ \begin{bmatrix} \underline{0}'_{k_i-1} & I_{k_i-1} \\ 0 & \underline{0}_{k_i-1} \end{bmatrix} \otimes \underline{B}'_{0,i}\underline{\beta}_i & \text{, if } k_i \geq 2. \end{cases}$$

Based on this notation, the diagonal components of $Q$ can be expressed as

$$Q_{m,m} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C_2-1 \\ C_2 \end{array} \begin{array}{c} \begin{array}{cccccc} 0 & 1 & 2 & \dots & C_2-1 & C_2 \end{array} \\ \left[ \begin{array}{cccccc} Q_{m,m,0} & (UD)_{m,0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ (LD)_{m,1} & Q_{m,m,1} & (UD)_{m,1} & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (LD)_{m,2} & Q_{m,m,2} & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & Q_{m,m,C_2-1} & (UD)_{m,C_2-1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & (LD)_{m,C_2} & Q_{m,m,C_2} \end{array} \right] \end{array},$$

where for $m = 0$

$$Q_{0,0,0} = -\lambda_{0,0} I_s + \left[ \begin{array}{cc} S_1 & \underline{S}'_{0,1}\underline{\gamma}_2 \\ \underline{S}'_{0,2}\underline{\gamma}_1 & S_2 \end{array} \right],$$

and

$$Q_{0,0,n} = \left[ \begin{array}{ccc} \zeta_{0,n,1} & \underline{S}'_{0,1}\underline{\gamma}_2 & \mathbf{0} \\ \mathbf{0} & \zeta_{0,n,3} & \underline{e}_{k_2,1} \otimes \left( \underline{S}'_{0,2}\underline{\beta}_2 \right) \\ \mathbf{0} & \mathbf{0} & \zeta_{0,n,4} \end{array} \right], \ n = 1, 2, \dots, C_2,$$

$$(UD)_{0,0} = \left[ \begin{array}{cc} \lambda_2 I_s & \underline{0}'_s \underline{0}_{k_2 b_2} \end{array} \right],$$

$$(UD)_{0,n} = \lambda_2 I_{z_1}, \ n = 1, 2, \dots, C_2 - 1,$$

$$(LD)_{0,1} = \left[ \begin{array}{cc} \alpha_2^{[1]} I_{s_1} & \mathbf{0} \\ \mathbf{0} & \alpha_2^{[1]} I_{s_2} \\ \underline{e}'_{k_2} \otimes \left( \underline{B}'_{0,2}\underline{\gamma}_1 \right) & \mathbf{0} \end{array} \right],$$

and

$$(LD)_{0,n} = \left[ \begin{array}{ccc} \alpha_2^{[n]} I_{s_1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \alpha_2^{[n]} I_{s_2} & \mathbf{0} \\ \underline{e}'_{k_2,k_2} \otimes \left( \underline{B}'_{0,2}\underline{\gamma}_1 \right) & \mathbf{0} & \alpha_2^{[n-1]} I_{k_2 b_2} + U_2 \end{array} \right], \ n = 2, 3, \dots, C_2.$$

For levels $m = 1, 2, \dots, C_1$,

$$Q_{m,m,0} = \left[ \begin{array}{ccc} \zeta_{m,0,1} & \underline{e}_{k_1,1} \otimes \left( \underline{S}'_{0,1}\underline{\beta}_1 \right) & \mathbf{0} \\ \mathbf{0} & \zeta_{m,0,2} & \mathbf{0} \\ \underline{S}'_{0,2}\underline{\gamma}_1 & \mathbf{0} & \zeta_{m,0,3} \end{array} \right],$$

and

$$Q_{m,m,n} = \left[ \begin{array}{cccc} \zeta_{m,n,1} & \underline{e}_{k_1,1} \otimes \left( \underline{S}'_{0,1}\underline{\beta}_1 \right) & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \zeta_{m,n,2} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \zeta_{m,n,3} & \underline{e}_{k_2,1} \otimes \left( \underline{S}'_{0,2}\underline{\beta}_2 \right) \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \zeta_{m,n,4} \end{array} \right], \ n = 1, 2, \dots, C_2,$$

while

$$(UD)_{m,0} = \left[ \begin{array}{cc} \lambda_2 I_{s+k_1 b_1} & \underline{0}'_{s+k_1 b_1} \underline{0}_{k_2 b_2} \end{array} \right],$$

$$(UD)_{m,n} = \lambda_2 I_{z_2},\ n = 1, 2, \ldots, C_2 - 1,$$

$$(LD)_{m,1} = \begin{bmatrix} \alpha_2^{[1]} I_{s_1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \alpha_2^{[1]} I_{k_1 b_1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \alpha_2^{[1]} I_{s_2} \\ \underline{e}'_{k_2} \otimes \left( \underline{B}'_{0,2} \underline{\gamma}_1 \right) & \mathbf{0} & \mathbf{0} \end{bmatrix},$$

and

$$(LD)_{m,n} = \begin{bmatrix} \alpha_2^{[n]} I_{s_1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \alpha_2^{[n]} I_{k_1 b_1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \alpha_2^{[n]} I_{s_2} & \mathbf{0} \\ \underline{e}'_{k_2,k_2} \otimes \left( \underline{B}'_{0,2} \underline{\gamma}_1 \right) & \mathbf{0} & \mathbf{0} & \alpha_2^{[n-1]} I_{k_2 b_2} + U_2 \end{bmatrix},\ n = 2, 3, \ldots, C_2.$$

The lower diagonal blocks of $Q$ have the form

$$Q_{m,m-1} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C_2 - 1 \\ C_2 \end{array} \begin{array}{c} \begin{array}{cccccc} 0 & 1 & 2 & \cdots & C_2 - 1 & C_2 \end{array} \\ \begin{bmatrix} Q_{m,m-1,0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & Q_{m,m-1,1} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & Q_{m,m-1,2} & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q_{m,m-1,C_2-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & Q_{m,m-1,C_2} \end{bmatrix} \end{array},$$

where for $m = 1$

$$Q_{1,0,0} = \begin{bmatrix} \alpha_1^{[1]} I_{s_1} & \mathbf{0} \\ \mathbf{0} & \underline{e}'_{k_1} \otimes \left( \underline{B}'_{0,1} \underline{\gamma}_2 \right) \\ \mathbf{0} & \alpha_1^{[1]} I_{s_2} \end{bmatrix},$$

and

$$Q_{1,0,n} = \begin{bmatrix} \alpha_1^{[1]} I_{s_1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \underline{e}'_{k_1} \otimes \left( \underline{B}'_{0,1} \underline{\gamma}_2 \right) & \mathbf{0} \\ \mathbf{0} & \alpha_1^{[1]} I_{s_2} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \alpha_1^{[1]} I_{k_2 b_2} \end{bmatrix},\ n = 1, 2, \ldots, C_2,$$

while for $m = 2, 3, \ldots, C_1$,

$$Q_{m,m-1,0} = \begin{bmatrix} \alpha_1^{[m]} I_{s_1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \alpha_1^{[m-1]} I_{k_1 b_1} + U_1 & \underline{e}'_{k_1,k_1} \otimes \left( \underline{B}'_{0,1} \underline{\gamma}_2 \right) \\ \mathbf{0} & \mathbf{0} & \alpha_1^{[m]} I_{s_2} \end{bmatrix},$$

and

$$Q_{m,m-1,n} = \begin{bmatrix} \alpha_1^{[m]} I_{s_1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \alpha_1^{[m-1]} I_{k_1 b_1} + U_1 & \underline{e}'_{k_1,k_1} \otimes \left( \underline{B}'_{0,1} \underline{\gamma}_2 \right) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \alpha_1^{[m]} I_{s_2} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \alpha_1^{[m]} I_{k_2 b_2} \end{bmatrix},\ n = 1, 2, \ldots, C_2.$$

Finally, the only remaining block to define is

$$Q_{0,1} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C_2 \end{array} \begin{array}{c} \begin{array}{ccccc} 0 & 1 & 2 & \dots & C_2 \end{array} \\ \left[ \begin{array}{ccccc} Q_{0,1,0} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & Q_{0,1,1} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & Q_{0,1,2} & \ddots & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & Q_{0,1,C_2} \end{array} \right] \end{array},$$

where

$$Q_{0,1,0} = \left[ \begin{array}{ccc} \lambda_1 I_{s_1} & \underline{0}'_{s_1}\underline{0}_{k_1b_1} & \mathbf{0} \\ \mathbf{0} & \underline{0}'_{s_2}\underline{0}_{k_1b_1} & \lambda_1 I_{s_2} \end{array} \right],$$

and

$$Q_{0,1,n} = \left[ \begin{array}{ccc} \lambda_1 I_{s_1} & \underline{0}'_{s_1}\underline{0}_{k_1b_1} & \mathbf{0} \\ \mathbf{0} & \underline{0}'_{s_2+k_2b_2}\underline{0}_{k_1b_1} & \lambda_1 I_{s_2+k_2b_2} \end{array} \right], \ n = 1, 2, \dots, C_2.$$

With the determination of these steady-state probabilities, we introduce two important quantities of interest associated with this particular queueing system. First of all, $P_{C_1,\bullet} = \sum_{j=0}^{C_2} P_{C_1,j}$ represents the probability that an arbitrarily arriving class-1 customer is turned away at entry (and subsequently lost) due to the class-1 queue being full, and is referred to as the *class-1 blocking probability.* Likewise, the *class-2 blocking probability* is given by $P_{\bullet,C_2} = \sum_{m=0}^{C_1} P_{m,C_2}$, and it represents the probability that an arbitrarily arriving class-2 customer is denied entry to the system due to the class-2 queue being full. We remark that $P_{C_1,\bullet}$ and $P_{\bullet,C_2}$ are particularly useful in helping choose values of $C_1$ and $C_2$ so as to ensure negligible blocking probabilities are obtained for both queues (should one want to use this model to approximate an infinite buffer system).

## 4.4 Determination of the Waiting Time Distribution

We derive the steady-state distribution of the random variable $W_i$, $i = 1, 2$, representing the duration of time from the (successful) arrival of an arbitrary class-$i$ customer to the system until the server is reached. For reasons that will become evident shortly, we refer to $W_i$ as the *nominal* class-$i$ waiting time. Without loss of generality, we focus our analysis only on $W_1$ as the characteristics of the two queues are essentially indifferent. In other words, the approach we develop below to obtain the distribution of $W_1$ can readily be adapted (via a simple relabeling of classes 1 and 2) to obtain the distribution of $W_2$.

First, recall that in a standard finite-buffer system, if a customer arrives to find their queue full, they are turned away and lost. Therefore, any potential arrivals when $X_1(t) = C_1$ are not observed. As in Equation (2.2), if we let $C_{1,h}$ denote the event of observing a class-1 customer arrival within the next $h$ time units and $S_{m,n,l,k,y}$ denote the event that $(X_1(t), X_2(t), L(t), K(t), Y(t)) = (m, n, l, k, y)$ at steady state (such that $P(S_{m,n,l,k,y}) = \pi_{m,n,l,k,y}$), then for $m < C_1$,

$$\begin{aligned}
q_{m,n,l,k,y} &= P((X_1(t), X_2(t), L(t), K(t), Y(t)) = (m, n, l, k, y) \text{ immediately prior to a class-1 arrival}) \\
&= \lim_{h\to 0} P(S_{m,n,l,k,y}|C_{1,h}) \\
&= \lim_{h\to 0} \frac{P(C_{1,h}|S_{m,n,l,k,y})P(S_{m,n,l,k,y})}{\sum_{x_1}\sum_{x_2}\sum_i\sum_j\sum_w P(C_{1,h}|S_{x_1,x_2,i,j,w})P(S_{x_1,x_2,i,j,w})} \\
&= \lim_{h\to 0} \frac{(\lambda_1 + o(h))\pi_{m,n,l,k,y}}{\sum_{x_1\neq C_1}\sum_{x_2}\sum_i\sum_j\sum_w (\lambda_1 + o(h))\pi_{x_1,x_2,i,j,w}} \\
&= \lim_{h\to 0} \frac{\lambda_1\pi_{m,n,l,k,y} + o(h)/h}{\lambda_1(1 - P_{C_1,\bullet}) + o(h)/h} \\
&= \frac{\pi_{m,n,l,k,y}}{1 - P_{C_1,\bullet}},
\end{aligned}$$

where in the fourth equality we remove the summation index of $x_1 = C_1$ since $P(C_{1,h}|S_{C_1,x_2,i,j,w}) = 0$, for all $x_2, i, j, w$. Hence, these are simply the steady-state probabilities, re-normalized by the fact that an observed class-1 arrival can not have been blocked. Correspondingly, define the re-normalized probability row vectors

$$\underline{\phi}_m = \frac{\underline{\pi}_m}{1 - P_{C_1,\bullet}}, \ m = 0, 1, 2, \ldots, C_1 - 1.$$

If we now construct

$$\underline{\Phi} = (\underline{\phi}_{C_1-1}, \underline{\phi}_{C_1-2}, \ldots, \underline{\phi}_1, \underline{\phi}_0) \tag{4.5}$$

to be the concatenated row vector of dimension

$$\ell = (C_1 - 1)n_2 + n_1, \tag{4.6}$$

then $\underline{\Phi}\,\underline{e}' = 1$ due to our earlier observation that, even when both queues are empty, the server is still busy in the midst of completing a switchover (and thus the wait time will be non-zero).

For the moment, we assume that our target class-1 customer is not subject to reneging (later on, we will incorporate the reneging behaviour of this specific customer back into the problem). While waiting in the class-1 queue, the number of customers in the class-2 queue potentially changes, not to mention the service indicator component used to identify how many customers have completed service within the active serving cycle. On the other hand, as the number of customers in the class-1 queue changes, the ones arriving later have no impact on the waiting time of the target class-1 customer. Therefore, if we effectively think of the arrival rate for the class-1 queue to be equal to 0, the distribution of $W_1$ can in fact be modelled as the distribution of the time to absorption in a Markov chain with infinitesimal generator of the form

$$\begin{bmatrix} \mathcal{R} & \underline{\mathcal{R}}_0' \\ \underline{0}_\ell & 0 \end{bmatrix},$$

where

$$\mathcal{R} = \begin{array}{c} \\ C_1-1 \\ C_1-2 \\ C_1-3 \\ \vdots \\ 2 \\ 1 \\ 0 \end{array} \begin{array}{c} \begin{array}{ccccccc} C_1-1 & C_1-2 & C_1-3 & \cdots & 2 & 1 & 0 \end{array} \\ \left[\begin{array}{ccccccc} \widetilde{Q}_{C_1-1,C_1-1} & Q_{C_1-1,C_1-2} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \widetilde{Q}_{C_1-2,C_1-2} & Q_{C_1-2,C_1-3} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \widetilde{Q}_{C_1-3,C_1-3} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \widetilde{Q}_{2,2} & Q_{2,1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \widetilde{Q}_{1,1} & \widetilde{Q}_{1,0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \widetilde{Q}_{0,0} \end{array}\right] \end{array} \qquad (4.7)$$

and $\underline{\mathcal{R}}'_0 = -\mathcal{R}\underline{e}'$. In Equation (4.7), the sub-matrices $Q_{2,1}, Q_{3,2}, \ldots, Q_{C_1-1,C_1-2}$ are identical to those defined in Section 4.3 and $\widetilde{Q}_{m,m} = Q_{m,m} + \lambda_1 I_{n_2}, m = 1, 2, \ldots, C_1 - 1$. Moreover, the levels $0, 1, \ldots, C_1 - 1$ of $\mathcal{R}$ represent how many possible customers are in the class-1 queue in front of our target customer upon arrival. Using the same notation from Section 4.3 whenever possible, it readily follows that

$$\tilde{Q}_{1,0} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C_2 \end{array} \begin{array}{c} \begin{array}{ccccc} 0 & 1 & 2 & \ldots & C_2 \end{array} \\ \left[\begin{array}{ccccc} \tilde{Q}_{1,0,0} & \mathbf{0} & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{0} & \tilde{Q}_{1,0,1} & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \tilde{Q}_{1,0,2} & \ldots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \ldots & \tilde{Q}_{1,0,C_2} \end{array}\right] \end{array},$$

where

$$\tilde{Q}_{1,0,0} = \left[\begin{array}{cc} \alpha_1^{[1]} I_{s_1} & \mathbf{0} \\ \mathbf{0} & \underline{e}'_{k_1,k_1} \otimes \left(\underline{B}'_{0,1}\underline{\gamma}_2\right) \\ \mathbf{0} & \alpha_1^{[1]} I_{s_2} \end{array}\right],$$

and

$$\tilde{Q}_{1,0,n} = \left[\begin{array}{ccc} \alpha_1^{[1]} I_{s_1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \underline{e}'_{k_1,k_1} \otimes \left(\underline{B}'_{0,1}\underline{\gamma}_2\right) & \mathbf{0} \\ \mathbf{0} & \alpha_1^{[1]} I_{s_2} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \alpha_1^{[1]} I_{k_2 b_2} \end{array}\right], \; n = 1, 2, \ldots, C_2,$$

and

$$\tilde{Q}_{0,0} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C_2-1 \\ C_2 \end{array} \begin{array}{c} \begin{array}{cccccc} 0 & 1 & 2 & \ldots & C_2-1 & C_2 \end{array} \\ \left[\begin{array}{cccccc} \tilde{Q}_{0,0,0} & (UD)_{0,0} & \mathbf{0} & \ldots & \mathbf{0} & \mathbf{0} \\ (LD)_{0,1} & \tilde{Q}_{0,0,1} & (UD)_{0,1} & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (LD)_{0,2} & \tilde{Q}_{0,0,2} & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \ldots & \tilde{Q}_{0,0,C_2-1} & (UD)_{0,C_2-1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \ldots & (LD)_{0,C_2} & \tilde{Q}_{0,0,C_2} \end{array}\right] \end{array},$$

where

$$\tilde{Q}_{0,0,0} = -\lambda_2 I_s + \begin{bmatrix} S_1 & \mathbf{0} \\ \underline{S}'_{0,2}\underline{\gamma}_1 & S_2 \end{bmatrix},$$

and

$$\tilde{Q}_{0,0,n} = \begin{bmatrix} \zeta_{0,n,1} + \lambda_1 I_{s_1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \zeta_{0,n,3} + \lambda_1 I_{s_2} & \underline{e}_{k_2,1} \otimes \left(\underline{S}'_{0,2}\underline{\beta}_2\right) \\ \mathbf{0} & \mathbf{0} & \zeta_{0,n,4} + \lambda_1 I_{k_2 b_2} \end{bmatrix}, \ n = 1, 2, \ldots, C_2,$$

for $n = 1, 2, \ldots, C_2$, and $(UD)_{0,n}$ and $(LD)_{0,n}$ are as previously defined.

According to the structure of the rate matrix $\mathcal{R}$, once our target customer enters the class-1 queue, the Markov chain will progressively make transitions from higher levels to lower ones, indicating the fact that the number of customers in front of the target customer reduces over time. The time to absorption is phase-type distributed with representation $\text{PH}_\ell(\underline{\Phi}, \mathcal{R})$, and so by Equation (1.11) we know that the distribution function of $W_1$, denoted by $F_1(\omega)$, is given by

$$F_1(\omega) = 1 - \underline{\Phi} \exp\{\mathcal{R}\omega\}\underline{e}', \, \omega \geq 0,$$

If we now proceed to include the reneging behaviour of our target class-1 customer by defining $W_1^*$ to be the *actual* class-1 waiting time (i.e., the arriving class-1 customer's total time spent in system prior to *successfully* entering service), then it clearly follows that

$$\begin{aligned} G_1(\omega) &= P(W_1^* < \omega) \\ &= 1 - P(W_1^* > \omega) \\ &= 1 - P(W_1^{\#} > \omega | \text{Reach Service}) \\ &= 1 - \frac{P(W_1^{\#} > \omega, \text{Reach Service})}{P(\text{Reach Service})}, \end{aligned}$$

where $W_1^{\#}$ is defined as the time that a class-1 customer spends in the system. We note that the reneging rate of the target customer depends on their position within the class-1 queue as well as the position of the server. Define the ordered vector $\underline{\alpha}_1$ to hold the target customer's individual reneging rate over all possible states of $\mathcal{R}$,

$$\begin{aligned} \underline{\alpha}_1 = (&\alpha_{1,C_1}\underline{e}_{s_1}, \alpha_{1,C_1-1}\underline{e}_{k_1 b_1}, \alpha_{1,C_1}\underline{e}_{s_2}, \text{rep}\{(\alpha_{1,C_1}\underline{e}_{s_1}, \alpha_{1,C_1-1}\underline{e}_{k_1 b_1}, \alpha_{1,C_1}\underline{e}_{s_2+k_2 b_2}), C_2\}, \\ &\alpha_{1,C_1-1}\underline{e}_{s_1}, \alpha_{1,C_1-2}\underline{e}_{k_1 b_1}, \alpha_{1,C_1-1}\underline{e}_{s_2}, \text{rep}\{(\alpha_{1,C_1-1}\underline{e}_{s_1}, \alpha_{1,C_1-2}\underline{e}_{k_1 b_1}, \alpha_{1,C_1-1}\underline{e}_{s_2+k_2 b_2}), C_2\}, \ldots, \\ &\alpha_{1,2}\underline{e}_{s_1}, \alpha_{1,1}\underline{e}_{k_1 b_1}, \alpha_{1,2}\underline{e}_{s_2}, \text{rep}\{(\alpha_{1,2}\underline{e}_{s_1}, \alpha_{1,1}\underline{e}_{k_1 b_1}, \alpha_{1,2}\underline{e}_{s_2+k_2 b_2}), C_2\}, \alpha_{1,1}\underline{e}_{n_1}), \end{aligned}$$

and let $\mathcal{A}_1 = \text{diag}(\underline{\alpha}_1)$ be the matrix with this vector as its main diagonal and zeroes everywhere else. It immediately follows that since $W_1^{\#}$ is the minimum of a customer's nominal waiting time and impatience time, we must have $W_1^{\#} \sim \text{PH}_\ell(\underline{\Phi}, \mathcal{R} - \mathcal{A}_1)$. We begin by deriving the probability of the target customer reaching service. If we consider a CTMC tracking the customer's progress through the system, with 'reneging' and 'reaching service' as two competing absorption states, the infinitesimal generator matrix for this process is

$$\begin{bmatrix} \mathcal{R} - \mathcal{A}_1 & \underline{\mathcal{R}}'_0 & \underline{\alpha}'_1 \\ \underline{0}_\ell & 0 & 0 \\ \underline{0}_\ell & 0 & 0 \end{bmatrix}, \tag{4.8}$$

with corresponding initial probability vector $(\underline{\Phi}, 0, 0)$. Therefore, absorption into the right-most state corresponds to the target reneging from the system, while absorption into the other absorbing state corresponds to reaching service. To determine the absorption probabilities for these two states, we apply Equation (A.17) from the Appendix. It is straightforward to confirm that

$$P(\text{Reach Service}) = \underline{\Phi}(\mathcal{A}_1 - \mathcal{R})^{-1}\underline{\mathcal{R}}'_0, \tag{4.9}$$

and

$$P(\text{Renege}) = \underline{\Phi}(\mathcal{A}_1 - \mathcal{R})^{-1}\underline{\alpha}'_1. \tag{4.10}$$

Next, letting $A_\omega$ denote the event that "the CTMC is in one of the $\ell$ transient states at time $\omega$", it follows that

$$\begin{aligned}
&P(W_1^{\#} > \omega, \text{Reach Service}) \\
&= P(A_\omega)P(\text{CTMC is eventually absorbed into the service state}|A_\omega) \\
&= \underline{\Phi}\exp\{(\mathcal{R} - \mathcal{A}_1)\omega\} \times (\mathcal{A}_1 - \mathcal{R})^{-1}\underline{\mathcal{R}}'_0 \\
&= \underline{\Phi}(\mathcal{A}_1 - \mathcal{R})^{-1}\exp\{(\mathcal{R} - \mathcal{A}_1)\omega\}\underline{\mathcal{R}}'_0, \ \omega \geq 0.
\end{aligned}$$

Thus, it is easy to see that

$$G_1(\omega) = 1 - \frac{\underline{\Phi}(\mathcal{A}_1 - \mathcal{R})^{-1}\exp\{(\mathcal{R} - \mathcal{A}_1)\omega\}\underline{\mathcal{R}}'_0}{\underline{\Phi}(\mathcal{A}_1 - \mathcal{R})^{-1}\underline{\mathcal{R}}'_0}, \ \omega \geq 0, \tag{4.11}$$

and so

$$\begin{aligned}
g_1(\omega) &= \frac{\partial}{\partial\omega}G_1(\omega) \\
&= \frac{\partial}{\partial\omega}\left(1 - \frac{\underline{\Phi}(\mathcal{A}_1 - \mathcal{R})^{-1}\exp\{(\mathcal{R} - \mathcal{A}_1)\omega\}\underline{\mathcal{R}}'_0}{\underline{\Phi}(\mathcal{A}_1 - \mathcal{R}_1)^{-1}\underline{\mathcal{R}}'_0}\right) \\
&= -\frac{\underline{\Phi}(\mathcal{A}_1 - \mathcal{R})^{-1}(\mathcal{R} - \mathcal{A}_1)\exp\{(\mathcal{R} - \mathcal{A}_1)\omega\}\underline{\mathcal{R}}'_0}{\underline{\Phi}(\mathcal{A}_1 - \mathcal{R})^{-1}\underline{\mathcal{R}}'_0} \\
&= \frac{\underline{\Phi}\exp\{(\mathcal{R} - \mathcal{A}_1)\omega\}\underline{\mathcal{R}}'_0}{\underline{\Phi}(\mathcal{A}_1 - \mathcal{R})^{-1}\underline{\mathcal{R}}'_0}, \ \omega > 0.
\end{aligned} \tag{4.12}$$

From the PDF $g_1(\omega)$, we can directly derive the $r$th moment of the actual waiting time,

$$\begin{aligned}
\mathrm{E}[W_1^{*r}] &= \int_0^\infty x^r \frac{\underline{\Phi}\exp\{(\mathcal{R} - \mathcal{A}_1)x\}\underline{\mathcal{R}}'_0}{\underline{\Phi}(\mathcal{A}_1 - \mathcal{R})^{-1}\underline{\mathcal{R}}'_0}dx \\
&= \int_0^\infty x^r \frac{\underline{\Phi}\exp\{(\mathcal{R} - \mathcal{A}_1)x\}(\mathcal{R}_1 - \mathcal{A}_1)(\mathcal{R} - \mathcal{A}_1)^{-1}(-\mathcal{R}\underline{e}')}{\underline{\Phi}(\mathcal{A}_1 - \mathcal{R})^{-1}\underline{\mathcal{R}}'_0}dx \\
&= \left(\frac{\underline{\Phi}}{\underline{\Phi}(\mathcal{A}_1 - \mathcal{R})^{-1}\underline{\mathcal{R}}'_0}\right)\left(-\int_0^\infty x^r\exp\{(\mathcal{R} - \mathcal{A}_1)x\}(\mathcal{R} - \mathcal{A}_1)dx\right)\left((\mathcal{R} - \mathcal{A}_1)^{-1}\mathcal{R}\underline{e}'\right) \\
&= \left(\frac{\underline{\Phi}}{\underline{\Phi}(\mathcal{A}_1 - \mathcal{R})^{-1}\underline{\mathcal{R}}'_0}\right)\left((-1)^r r!(\mathcal{R} - \mathcal{A}_1)^{-r}\right)\left((\mathcal{R} - \mathcal{A}_1)^{-1}\mathcal{R}\underline{e}'\right) \\
&= \left(\frac{\underline{\Phi}}{\underline{\Phi}(\mathcal{A}_1 - \mathcal{R})^{-1}\underline{\mathcal{R}}'_0}\right)r!(\mathcal{A}_1 - \mathcal{R})^{-(r+1)}(-\mathcal{R}\underline{e}') \\
&= \frac{r!\underline{\Phi}(\mathcal{A}_1 - \mathcal{R})^{-(r+1)}\underline{\mathcal{R}}'_0}{\underline{\Phi}(\mathcal{A}_1 - \mathcal{R})^{-1}\underline{\mathcal{R}}'_0}, \ r = 1, 2, \ldots,
\end{aligned} \tag{4.13}$$

where we may pass the integration through the matrix products due to there being finite-many states. Note that the identity used in the fourth equality holds by Equation (1.14), since we know that for a $\text{PH}_M(\underline{\alpha}_0^*, S)$ random variable $X$,

$$\text{E}[X^r] = (-1)^r r! \underline{\alpha}_0^* S^{-r} \underline{e}' = \underline{\alpha}_0^* \left( (-1)^r r! S^{-r} \right) \underline{e}',$$

which we can alternately express as

$$\text{E}[X^r] = \int_0^\infty x^r \underline{\alpha}_0^* \exp\{Sx\} \underline{S}_0' dx = \underline{\alpha}_0^* \left( - \int_0^\infty x^r \exp\{Sx\} S dx \right) \underline{e}'.$$

Equating the two equations and replacing $S$ by $\mathcal{R} - \mathcal{A}_1$ results in the required identity. Finally, we end with a reminder that the corresponding results for $W_2^*$ can be obtained in a completely analogous fashion.

**Remark 4.1.** If we had the case that $\alpha_{1,n} = \alpha_1$, $n = 1, 2, \ldots, C_1$, then $\underline{\alpha}_1 = \alpha_1 \underline{e}_\ell$ and $\mathcal{A}_1 = \alpha_1 I_\ell$. This implies that $\mathcal{A}_1 \mathcal{R} = \mathcal{R} \mathcal{A}_1$ and so $\exp\{(\mathcal{R} - \mathcal{A}_1)\omega\} = \exp\{\mathcal{R}\omega\} \exp\{-\mathcal{A}_1 \omega\}$, as previously proven in Section 1.2.6. Next, it follows that

$$\exp\{-\mathcal{A}_1 \omega\} = \sum_{n=0}^\infty \frac{(-\omega)^n}{n!} \mathcal{A}_1^n = \sum_{n=0}^\infty \frac{(-\alpha_1 \omega)^n}{n!} I^n = e^{-\alpha_1 \omega} I_\ell,$$

and substitution into Equation (4.11) yields

$$\begin{aligned} G_1(\omega) &= 1 - \frac{\underline{\Phi}(\alpha_1 I_\ell - \mathcal{R})^{-1} \exp\{\mathcal{R}\omega\} e^{-\alpha_1 \omega} (-\mathcal{R}\underline{e}')}{\underline{\Phi}(\alpha_1 I_\ell - \mathcal{R})^{-1} \underline{\mathcal{R}}_0'} \\ &= 1 - \frac{\underline{\Phi}(\alpha_1 I_\ell - \mathcal{R})^{-1} (-\mathcal{R}) \exp\{\mathcal{R}\omega\} e^{-\alpha_1 \omega} \underline{e}'}{\underline{\Phi}(\alpha_1 I_\ell - \mathcal{R})^{-1} \underline{\mathcal{R}}_0'} \\ &= 1 - \frac{\underline{\Phi}(\alpha_1 I_\ell - \mathcal{R})^{-1} (-\mathcal{R})}{\underline{\Phi}(\alpha_1 I_\ell - \mathcal{R})^{-1} (-\mathcal{R}\underline{e}')} \exp\{(\mathcal{R} - \alpha_1 I_\ell)\omega\} \underline{e}', \ \omega \geq 0, \end{aligned}$$

which we can recognize as the CDF of a continuous phase-type distribution, and so

$$W_1^* \sim \text{PH}_\ell \left( \frac{\underline{\Phi}(\alpha_1 I_\ell - \mathcal{R})^{-1} (-\mathcal{R})}{\underline{\Phi}(\alpha_1 I_\ell - \mathcal{R})^{-1} (-\mathcal{R}) \underline{e}'}, \mathcal{R} - \alpha_1 I_\ell \right).$$

From the theory of phase-type distributions, this immediately tells us that

$$\begin{aligned} g_1(\omega) &= \frac{\underline{\Phi}(\alpha_1 I_\ell - \mathcal{R})^{-1} (-\mathcal{R})}{\underline{\Phi}(\alpha_1 I_\ell - \mathcal{R})^{-1} (-\mathcal{R}) \underline{e}'} \exp\{(\mathcal{R} - \alpha_1 I_\ell)\omega\} (-(\mathcal{R} - \alpha_1 I_\ell) \underline{e}') \\ &= \frac{\underline{\Phi}(\alpha_1 I_\ell - \mathcal{R})^{-1} (-(\mathcal{R} - \alpha_1 I_\ell))}{\underline{\Phi}(\alpha_1 I_\ell - \mathcal{R})^{-1} (-\mathcal{R}) \underline{e}'} \exp\{(\mathcal{R} - \alpha_1 I_\ell)\omega\} (-\mathcal{R}\underline{e}') \\ &= \frac{\underline{\Phi} \exp\{(\mathcal{R} - \alpha_1 I_\ell)\omega\} \underline{\mathcal{R}}_0'}{\underline{\Phi}(\alpha_1 I_\ell - \mathcal{R})^{-1} \underline{\mathcal{R}}_0'}, \ \omega > 0, \end{aligned}$$

agreeing with Equation (4.12), where the second equality holds since $\mathcal{R}(\mathcal{R} - \alpha_1 I_\ell) = (\mathcal{R} - \alpha_1 I_\ell)\mathcal{R}$, and

$$\begin{aligned} \text{E}[W_1^{*r}] &= (-1)^r r! \frac{\underline{\Phi}(\alpha_1 I_\ell - \mathcal{R})^{-1} (-\mathcal{R})}{\underline{\Phi}(\alpha_1 I_\ell - \mathcal{R})^{-1} (-\mathcal{R}) \underline{e}'} (\mathcal{R} - \alpha_1 I_\ell)^{-r} \underline{e}' \\ &= \frac{(-1)^r r! \underline{\Phi}(\alpha_1 I_\ell - \mathcal{R})^{-1} (\mathcal{R} - \alpha_1 I_\ell)^{-r} (-\mathcal{R}\underline{e}')}{\underline{\Phi}(\alpha_1 I_\ell - \mathcal{R})^{-1} (-\mathcal{R}\underline{e}')} \\ &= \frac{r! \underline{\Phi}(\alpha_1 I_\ell - \mathcal{R})^{-(r+1)} \underline{\mathcal{R}}_0'}{\underline{\Phi}(\alpha_1 I_\ell - \mathcal{R})^{-1} \underline{\mathcal{R}}_0'}, \ r = 1, 2, \ldots, \end{aligned} \tag{4.14}$$

agreeing with Equation (4.13).

**Remark 4.2.** The analysis within this subsection assumes that we are treating the system as having an actual finite buffer, rather than trying to approximate an infinite buffer model as accurately as possible. If we wanted the latter, then we could improve the accuracy by allowing the target class-1 customer to stay in the system if they observe $X_1 = C_1$. In practice, this can be accomplished by not re-normalizing the steady-state probabilities, letting the initial probability row vector be

$$\underline{\Phi} = (\underline{\pi}_{C_1}, \underline{\pi}_{C_1-1}, \dots, \underline{\pi}_1, \underline{\pi}_0),$$

which would now have length $C_1 n_2 + n_1$ (while still satisfying $\underline{\Phi}\,\underline{e}'_{C_1 n_2 + n_1} = 1$), and appropriately adjusting Equation (4.7) to include a row and column of blocks for level $C_1$.

## 4.5 Numerical Examples

### 4.5.1 A Cost Optimization Problem Concerning Expected Time Spent Waiting in System

In this subsection, we investigate the selection of service discipline parameters $k_1$ and $k_2$ in order to optimize the system by way of minimizing a particular cost function. This cost function is from the system's point-of-view, and so we are concerned with the time a customer spends in the system, $W_i^{\#}$, rather than their actual or nominal waiting times. Recall that $W_i^{\#} \sim \mathrm{PH}_{\ell_i}(\underline{\Phi}_i, \mathcal{R}_i - \mathcal{A}_i)$, so we know that

$$\mathrm{E}[W_i^{\#}] = \underline{\Phi}_i(\mathcal{A}_i - \mathcal{R}_i)^{-1}\underline{e}', \ i = 1, 2,$$

where $\underline{\Phi}_1$, $\ell_1$, and $\mathcal{R}_1$ are given by Equations (4.5), (4.6), and (4.7), respectively, and $\mathcal{A}_2$, $\underline{\Phi}_2$, $\ell_2$, and $\mathcal{R}_2$ are similarly determined.

In what follows, we consider the cost function given by

$$\mathrm{Cost} = \mathrm{Cost}_1 + \mathrm{Cost}_2,$$

where

$$\mathrm{Cost}_i = c_i \lambda_i \mathrm{E}[W_i^{\#}] + r_i \lambda_i \mathrm{Pr}(\text{Class-}i \text{ customer reneges}), \ i = 1, 2,$$

and $c_i$ and $r_i$ are assumed to be non-negative constants representing the waiting cost parameter associated with class $i$ and the penalty cost parameter associated with a class-$i$ customer who reneges, respectively. From Equation (4.10), the probability of a class-$i$ customer reneging is $\underline{\Phi}_i(\mathcal{A}_i - \mathcal{R}_i)^{-1}\underline{\alpha}'_i$. Therefore, we can ultimately show that the cost function for class-$i$ takes the form

$$\begin{aligned} \mathrm{Cost}_i &= \lambda_i c_i \underline{\Phi}_i(\mathcal{A}_i - \mathcal{R}_i)^{-1}\underline{e}' + \lambda_i r_i \underline{\Phi}_i(\mathcal{A}_i - \mathcal{R}_i)^{-1}\underline{\alpha}'_i \\ &= \lambda_i \underline{\Phi}_i(\mathcal{A}_i - \mathcal{R}_i)^{-1}(c_i \underline{e}' + r_i \underline{\alpha}'_i). \end{aligned} \tag{4.15}$$

For the remainder of this chapter, we consider the simplified version of the model where $\alpha_{i,n} = \alpha_i$, and so the individual reneging rates of each customer simply depends on their class, and nothing else. We will briefly visualize the impact of level-dependent reneging in Section 6.6.1, within the context of a further generalized model. In this simplified case, Equation (4.15) reduces to

$$\mathrm{Cost}_i = \lambda_i (c_i + r_i \alpha_i) \underline{\Phi}_i (\alpha_i I_{\ell_i} - \mathcal{R}_i)^{-1}\underline{e}'. \tag{4.16}$$

We remark that this choice of cost function is inspired by the work of Borst et al. [17], in which the authors studied a cyclic polling model with infinite buffers (but no reneging), and sought to determine optimal $k_i$ values so as to minimize the mean waiting cost of customers, subject to a constraint limiting the number of services per cycle. In particular, by setting the reneging rates $\alpha_1$ and $\alpha_2$ both equal to zero, our cost function reduces to their waiting cost function. Moreover, as a means of testing the accuracy of our results, we were able to replicate the choices of optimal $(k_1, k_2)$ in Table I.a, p. 607, of Borst et al. [17] by setting $\alpha_1 = \alpha_2 = 0$, choosing $C_1 = 29$ and $C_2 = 48$, and calculating the cost function for all $k_1 = 1, 2, \ldots, 11$, $k_2 = 1, \ldots, 12 - k_1$. These buffer sizes yielded blocking probabilities no larger than 0.002295 for class 1 and 0.023388 for class 2, which occurred in the most extreme combinations of $(k_1, k_2)$ – namely, $(1, 11)$ or $(11, 1)$.

Similar to the study conducted by Borst et al. [17], we investigate the behaviour of our proposed cost function and how optimal $(k_1, k_2)$ combinations might change in the presence of reneging and varying service time distributions, subject to the constraint $k_1 + k_2 \leq K$ which limits the number of services per cycle. As a point of comparison, we consider two specific parametric cases which are both drawn from Section IV of Borst et al. [17] with $K = 12$. In Case 1, we assume equal arrival rates $\lambda_1 = \lambda_2 = 0.75$, exponentially distributed switch-in times with equal rates $S_1 = S_2 = -1/0.1$, and mean service times of $\mu_1 = 0.9$ and $\mu_2 = 0.1$. In Case 2, we assume $\mu_1 = \mu_2 = 1$, along with differing arrival rates $\lambda_1 = 0.5$ and $\lambda_2 = 0.25$, and exponentially distributed switch-in times with $S_1 = -1/0.2$ and $S_2 = -1/0.1$. In both cases, we consider reneging rates $\alpha_1$ and $\alpha_2$ chosen from the set $\{0.025, 0.05, 0.25\}$. Furthermore, the distribution of class-$i$ service times could be one of the following:

- (Exp) Exponential: $\mathrm{E}[Ser_i] = \mu_i$ and $\mathrm{Var}(Ser_i) = \mu_i^2$:

$$Ser_i \sim \mathrm{PH}_1\left(\underline{\beta}_i = 1, B_i = -1/\mu_i\right).$$

- (H$_2$) Hyperexponential-2: $\mathrm{E}[Ser_i] = \mu_i$ and $\mathrm{Var}(Ser_i) = 1000\mu_i^2$:

$$Ser_i \sim \mathrm{PH}_2\left(\underline{\beta}_i = (0.001, 0.999), B_i = \begin{bmatrix} -\left(\frac{1}{\mu_i}\right)\left(\frac{\sqrt{2}}{\sqrt{2}+999}\right) & 0 \\ 0 & -\left(\frac{1}{\mu_i}\right)\left(\frac{\sqrt{2}}{\sqrt{2}-1}\right) \end{bmatrix}\right).$$

- (E$_3$) Erlang-3: $\mathrm{E}[Ser_i] = \mu_i$ and $\mathrm{Var}(Ser_i) = \mu_i^2/3$:

$$Ser_i \sim \mathrm{PH}_3\left(\underline{\beta}_i = (1, 0, 0), B_i = \begin{bmatrix} -3/\mu_i & 3/\mu_i & 0 \\ 0 & -3/\mu_i & 3/\mu_i \\ 0 & 0 & -3/\mu_i \end{bmatrix}\right).$$

The various parameter combinations resulted in a range of observed blocking probabilities, and the maximum blocking probability per class (over the different possible pairs of $k_1$ and $k_2$) for each combination of reneging rate and service time distribution was compared. Of these local maxima, class-1 blocking probabilities under Case 1 (Case 2) had a median of $8.431 \times 10^{-6}$ $(1.741 \times 10^{-6})$ and a global maximum of 0.1312 (0.0577). With respect to class 2, the local maxima under Case 1 (Case 2) possessed a median of $2.518 \times 10^{-4}$ $(2.772 \times 10^{-10})$ and a global maximum of 0.1242 (0.00075). Although our model, with buffer sizes of $C_1 = C_2 = 20$ used throughout, falls short at emulating (with high accuracy) the corresponding infinite buffer

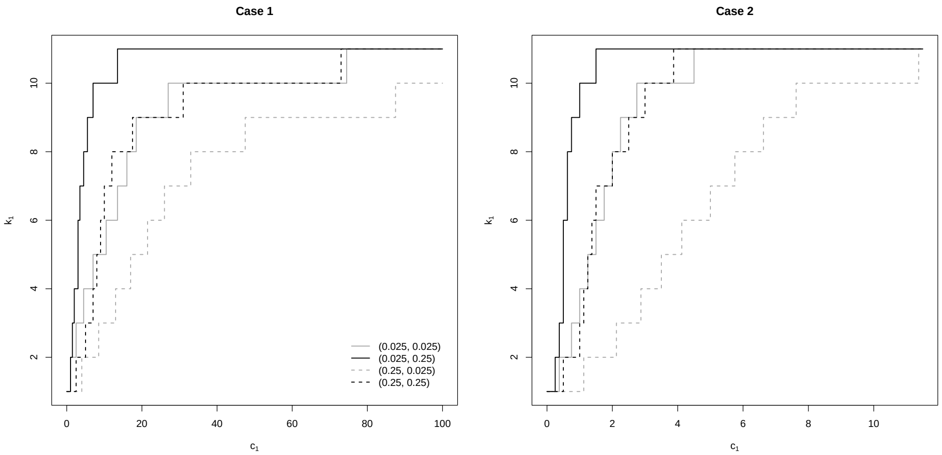

Figure 4.2 (a)

Figure 4.2 (b)

Figure 4.2: Plots of $k_1$ versus $c_1$ under both Cases 1 and 2 with Exp service times, $c_2 = 2$, $r_1 = r_2 = 1$, and four combinations of reneging rates.

system for a few combinations of $k_i$, $\alpha_i$, and service time distribution (particularly in situations involving the variance-inflated $H_2$ service time distributions and low reneging rates), it does a more than adequate job when using only Exp or $E_3$ service, or when reneging rates are high. If the goal is to precisely emulate an infinite buffer system under those aforementioned conditions (e.g., extremely large service time variance), we would recommend increasing $C_1$ and $C_2$, computational resources permitting, to achieve more tolerable blocking probabilities across all $k_i$ combinations. In an effort to keep computation times manageable, however, we elected to accept these blocking probabilities and use buffer sizes of 20 apiece over all parameter combinations.

Tables 4.1 and 4.2 display the optimal $(k_1, k_2)$ pairs, along with their corresponding cost values, for each combination of reneging rate and service time distribution under Cases 1 and 2, respectively. In each table, we present results corresponding to $c_1 = 2$, $c_2 = 1$, $r_1 = 1$, and $r_2 = 0.5$, as well as results for select combinations of service time distribution when $r_1 = r_2 = 40$. In looking at the optimal values of $k_1$ and $k_2$ under Case 1 over a range of cost parameters, we observed that the limit of the optimal choice of $(k_1, k_2)$ is $(11, 1)$ as $c_1$ or $r_1$ approaches $\infty$, or $(1, 11)$ as $c_2$ or $r_2$ approaches $\infty$. An example of this convergence is illustrated in Fig. 4.2, where we plotted the optimal values of $k_1$ against $c_1$ (with $c_2$, $r_1$, and $r_2$ held constant). The rates of convergence (to $k_1 = 11$) appear to be largely dependent on the relative values of $\alpha_1$ and $\alpha_2$. Note that $k_1$ converges faster when class-2 customers are more impatient, causing fewer of them to reach service and resulting in relatively longer class-1 queues. This causes class 1 to dominate the expected time waiting in system portion of the cost function, whereas class-2 customers dominate the probability of reneging portion. Since we are plotting against the class-1 waiting cost parameter (while keeping reneging costs constant), it is easy to see why the $(0.025, 0.25)$ combination converges the fastest and $(0.25, 0.025)$ the slowest, whereas equal reneging rate combinations tend to be comparable to one another. This result is consistent between Cases 1 and 2. In addition, note that the class-1 arrival rate is twice that of class 2 in Case 2, which results in costs associated with class 1 dominating the cost function sooner. As a result, we observed that Case 2's system converges to $(11, 1)$ faster and $(1, 11)$ slower in comparison to Case 1's system.

Table 4.1: Optimal $(k_1, k_2)$ and minimum cost values for Case 1 with $c_1 = 2$, $c_2 = 1$, and $r_1 = 1$, $r_2 = 0.5$ or $r_1 = r_2 = 40$.

| Reneging | | Service Time Distributions | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Rates | | (Exp, Exp) | | (Exp, $H_2$) | | (Exp, $E_3$) | | (Exp, Exp) | |
| $\alpha_1$ | $\alpha_2$ | $(k_1,\ k_2)$ | Cost | $(k_1,\ k_2)$ | Cost | $(k_1,\ k_2)$ | Cost | $(k_1,\ k_2)$ | Cost |
| 0.025 | 0.025 | (3, 9) | 4.3398 | (3, 9) | 6.2866 | (3, 9) | 4.3281 | (3, 9) | 6.9361 |
| | 0.05 | (4, 8) | 4.2581 | (3, 9) | 5.9977 | (4, 8) | 4.2468 | (2, 10) | 7.7429 |
| | 0.25 | (7, 5) | 3.7352 | (9, 3) | 4.8325 | (7, 5) | 3.7269 | (2, 10) | 11.9875 |
| 0.05 | 0.025 | (3, 9) | 3.6482 | (3, 9) | 5.2460 | (3, 9) | 3.6386 | (3, 9) | 7.1422 |
| | 0.05 | (3, 9) | 3.5947 | (3, 9) | 4.9824 | (3, 9) | 3.5855 | (2, 10) | 7.8847 |
| | 0.25 | (6, 6) | 3.2543 | (6, 6) | 4.0882 | (6, 6) | 3.2470 | (1, 11) | 11.9519 |
| 0.25 | 0.025 | (2, 10) | 2.1520 | (2, 10) | 3.0167 | (2, 10) | 2.1464 | (3, 9) | 9.0264 |
| | 0.05 | (2, 10) | 2.1334 | (2, 10) | 2.8169 | (2, 10) | 2.1279 | (3, 9) | 9.7272 |
| | 0.25 | (2, 10) | 2.0230 | (2, 10) | 2.2667 | (2, 10) | 2.0183 | (1, 11) | 13.3357 |
| | | ($H_2$, Exp) | | ($H_2$, $H_2$) | | ($H_2$, $E_3$) | | ($H_2$, $H_2$) | |
| $\alpha_1$ | $\alpha_2$ | $(k_1,\ k_2)$ | Cost | $(k_1,\ k_2)$ | Cost | $(k_1,\ k_2)$ | Cost | $(k_1,\ k_2)$ | Cost |
| 0.025 | 0.025 | (4, 8) | 20.3486 | (3, 9) | 21.7547 | (4, 8) | 20.3441 | (3, 9) | 35.8657 |
| | 0.05 | (5, 7) | 18.2711 | (4, 8) | 19.5065 | (5, 7) | 18.2666 | (3, 9) | 36.3322 |
| | 0.25 | (8, 4) | 14.6158 | (9, 3) | 15.5758 | (8, 4) | 14.6114 | (2, 10) | 33.8738 |
| 0.05 | 0.025 | (3, 9) | 16.4205 | (2, 10) | 17.4343 | (3, 9) | 16.4171 | (2, 10) | 34.1935 |
| | 0.05 | (4, 8) | 14.3710 | (3, 9) | 15.2235 | (4, 8) | 14.3676 | (3, 9) | 34.6786 |
| | 0.25 | (7, 5) | 10.7526 | (8, 4) | 11.3615 | (7, 5) | 10.7490 | (2, 10) | 32.2619 |
| 0.25 | 0.025 | (1, 11) | 8.8681 | (1, 11) | 9.4376 | (1, 11) | 8.8681 | (3, 9) | 27.1216 |
| | 0.05 | (1, 11) | 6.9769 | (1, 11) | 7.3890 | (1, 11) | 6.9756 | (3, 9) | 27.6632 |
| | 0.25 | (6, 6) | 3.5293 | (5, 7) | 3.7245 | (6, 6) | 3.5263 | (2, 10) | 25.4136 |
| | | ($E_3$, Exp) | | ($E_3$, $H_2$) | | ($E_3$, $E_3$) | | ($E_3$, $E_3$) | |
| $\alpha_1$ | $\alpha_2$ | $(k_1,\ k_2)$ | Cost | $(k_1,\ k_2)$ | Cost | $(k_1,\ k_2)$ | Cost | $(k_1,\ k_2)$ | Cost |
| 0.025 | 0.025 | (4, 8) | 3.4530 | (3, 9) | 5.5164 | (4, 8) | 3.4401 | (3, 9) | 5.5277 |
| | 0.05 | (4, 8) | 3.3965 | (4, 8) | 5.2494 | (4, 8) | 3.3839 | (3, 9) | 6.2533 |
| | 0.25 | (7, 5) | 3.0303 | (8, 4) | 4.2083 | (7, 5) | 3.0210 | (2, 10) | 10.1034 |
| 0.05 | 0.025 | (3, 9) | 2.9907 | (3, 9) | 4.6552 | (3, 9) | 2.9800 | (3, 9) | 5.8344 |
| | 0.05 | (3, 9) | 2.9587 | (3, 9) | 4.4143 | (3, 9) | 2.9483 | (2, 10) | 6.5219 |
| | 0.25 | (5, 7) | 2.7179 | (6, 6) | 3.6040 | (5, 7) | 2.7096 | (1, 11) | 10.1745 |
| 0.25 | 0.025 | (2, 10) | 1.8771 | (2, 10) | 2.7582 | (2, 10) | 1.8710 | (4, 8) | 7.8647 |
| | 0.05 | (2, 10) | 1.8677 | (2, 10) | 2.5692 | (2, 10) | 1.8618 | (3, 9) | 8.4997 |
| | 0.25 | (2, 10) | 1.8065 | (2, 10) | 2.0618 | (2, 10) | 1.8013 | (1, 11) | 11.8951 |
| $(r_1,\ r_2)$ | | (1, 0.5) | | (1, 0.5) | | (1, 0.5) | | (40, 40) | |

Table 4.2: Optimal $(k_1, k_2)$ and minimum cost values for Case 2 with $c_1 = 2$, $c_2 = 1$, and $r_1 = 1$, $r_2 = 0.5$ or $r_1 = r_2 = 40$.

| Reneging | | Service Time Distributions | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Rates | | (Exp, Exp) | | (Exp, $H_2$) | | (Exp, $E_3$) | | (Exp, Exp) | |
| $\alpha_1$ | $\alpha_2$ | $(k_1,\ k_2)$ | Cost | $(k_1,\ k_2)$ | Cost | $(k_1,\ k_2)$ | Cost | $(k_1,\ k_2)$ | Cost |
| 0.025 | 0.025 | (10, 2) | 2.6649 | (10, 2) | 7.8090 | (10, 2) | 2.4761 | (10, 2) | 4.3972 |
| | 0.05 | (11, 1) | 2.4083 | (11, 1) | 6.8184 | (11, 1) | 2.2577 | (9, 3) | 4.6890 |
| | 0.25 | (11, 1) | 1.8357 | (11, 1) | 5.1854 | (11, 1) | 1.7432 | (8, 4) | 5.9054 |
| 0.05 | 0.025 | (10, 2) | 2.3812 | (9, 3) | 5.6575 | (10, 2) | 2.2247 | (10, 2) | 4.6660 |
| | 0.05 | (10, 2) | 2.2030 | (10, 2) | 4.8100 | (10, 2) | 2.0669 | (10, 2) | 4.9625 |
| | 0.25 | (11, 1) | 1.6934 | (11, 1) | 3.5890 | (11, 1) | 1.6127 | (8, 4) | 6.1953 |
| 0.25 | 0.025 | (6, 6) | 1.5047 | (5, 7) | 2.9877 | (6, 6) | 1.4353 | (11, 1) | 6.4562 |
| | 0.05 | (7, 5) | 1.4550 | (6, 6) | 2.2615 | (7, 5) | 1.3907 | (10, 2) | 6.7245 |
| | 0.25 | (11, 1) | 1.2323 | (10, 2) | 1.5454 | (11, 1) | 1.1879 | (8, 4) | 7.8861 |
| | | ($H_2$, Exp) | | ($H_2$, $H_2$) | | ($H_2$, $E_3$) | | ($H_2$, $H_2$) | |
| $\alpha_1$ | $\alpha_2$ | $(k_1,\ k_2)$ | Cost | $(k_1,\ k_2)$ | Cost | $(k_1,\ k_2)$ | Cost | $(k_1,\ k_2)$ | Cost |
| 0.025 | 0.025 | (11, 1) | 11.9085 | (10, 2) | 15.6149 | (11, 1) | 11.8306 | (10, 2) | 24.8349 |
| | 0.05 | (11, 1) | 10.6576 | (11, 1) | 13.9119 | (11, 1) | 10.5843 | (10, 2) | 23.2350 |
| | 0.25 | (11, 1) | 9.5631 | (11, 1) | 12.3096 | (11, 1) | 9.5085 | (9, 3) | 21.9772 |
| 0.05 | 0.025 | (10, 2) | 8.2620 | (9, 3) | 10.5575 | (10, 2) | 8.1927 | (9, 3) | 20.6821 |
| | 0.05 | (11, 1) | 7.0367 | (10, 2) | 8.9056 | (11, 1) | 6.9730 | (9, 3) | 19.1262 |
| | 0.25 | (11, 1) | 5.9644 | (11, 1) | 7.4260 | (11, 1) | 5.9166 | (8, 4) | 17.9732 |
| 0.25 | 0.025 | (4, 8) | 3.9129 | (4, 8) | 5.0239 | (4, 8) | 3.8846 | (8, 4) | 15.7298 |
| | 0.05 | (9, 3) | 2.8207 | (6, 6) | 3.4547 | (9, 3) | 2.7818 | (8, 4) | 14.2528 |
| | 0.25 | (11, 1) | 1.8533 | (9, 3) | 2.1237 | (11, 1) | 1.8198 | (7, 5) | 13.2301 |
| | | ($E_3$, Exp) | | ($E_3$, $H_2$) | | ($E_3$, $E_3$) | | ($E_3$, $E_3$) | |
| $\alpha_1$ | $\alpha_2$ | $(k_1,\ k_2)$ | Cost | $(k_1,\ k_2)$ | Cost | $(k_1,\ k_2)$ | Cost | $(k_1,\ k_2)$ | Cost |
| 0.025 | 0.025 | (10, 2) | 2.2940 | (10, 2) | 7.5463 | (10, 2) | 2.0953 | (10, 2) | 3.4708 |
| | 0.05 | (11, 1) | 2.0828 | (11, 1) | 6.5993 | (11, 1) | 1.9246 | (9, 3) | 3.7560 |
| | 0.25 | (11, 1) | 1.5665 | (11, 1) | 5.0376 | (11, 1) | 1.4696 | (7, 5) | 5.0495 |
| 0.05 | 0.025 | (10, 2) | 2.0877 | (9, 3) | 5.4309 | (10, 2) | 1.9226 | (10, 2) | 3.7678 |
| | 0.05 | (10, 2) | 1.9375 | (10, 2) | 4.6075 | (10, 2) | 1.7938 | (9, 3) | 4.0603 |
| | 0.25 | (11, 1) | 1.4762 | (11, 1) | 3.4371 | (11, 1) | 1.3913 | (7, 5) | 5.3424 |
| 0.25 | 0.025 | (6, 6) | 1.3829 | (5, 7) | 2.8826 | (6, 6) | 1.3106 | (11, 1) | 5.6070 |
| | 0.05 | (7, 5) | 1.3415 | (7, 5) | 2.1606 | (7, 5) | 1.2744 | (10, 2) | 5.8895 |
| | 0.25 | (11, 1) | 1.1422 | (10, 2) | 1.4620 | (11, 1) | 1.0960 | (8, 4) | 7.0570 |
| $(r_1,\ r_2)$ | | (1, 0.5) | | (1, 0.5) | | (1, 0.5) | | (40, 40) | |

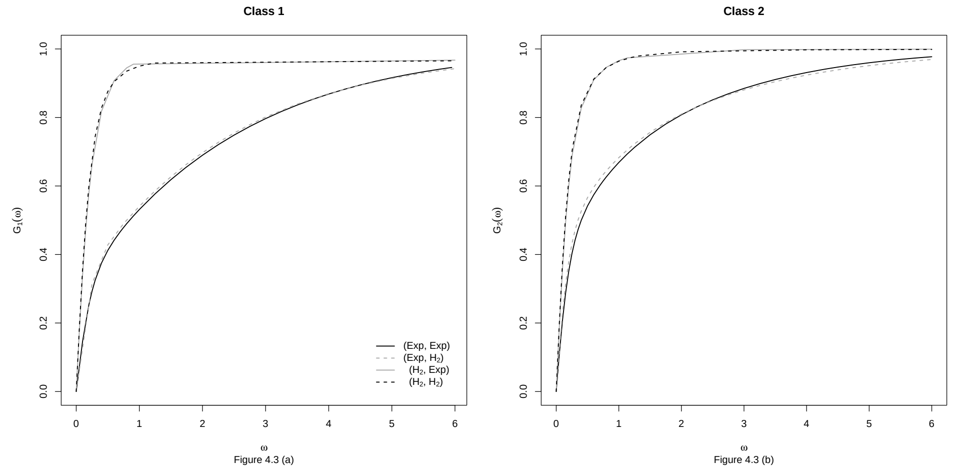

Figure 4.3: Plots of $G_i(\omega)$ versus $\omega$ for both classes under Case 1 with $\alpha_1 = 0.025$, $\alpha_2 = 0.25$, either Exp or H$_2$ service times, and optimal $(k_1, k_2)$ from Table 4.1.

Tables 4.1 and 4.2 also suggest that when the waiting and reneging cost parameters are of a comparable size (or the waiting cost parameters are much larger), the size of $k_i$ is inversely proportional to $\alpha_i$ (while keeping the other class's reneging rate constant). When the reneging cost parameters are much larger than the waiting cost parameters, such as in the $r_1 = r_2 = 40$ examples, this relationship may invert, as serving fewer class-$i$ customers per cycle, in combination with a larger $\alpha_i$ which increases the probability of a class-$i$ customer reneging before service becomes available, becomes more costly. In general, the system appears to be more sensitive to smaller changes in the waiting cost parameters. This is an intuitive result, as $r_i$ is multiplied by $\alpha_i$ in the cost function in Equation (4.16), which may be very small. Depending on the choice of reneging rates, this may lead to the cost contributed by reneging being much smaller than the cost due to customers waiting in the system.

For a given pair of $\alpha_1$ and $\alpha_2$, we observe that changing the service time distribution can impact the optimal choice of $k_1$ and $k_2$. Our results in Tables 4.1 and 4.2 indicate that it is possible to vary the optimal $(k_1, k_2)$ values by switching only one (or both) of the service time distributions. The larger the difference in variance between two service time distributions, the more likely we are to observe changes in the optimal $(k_1, k_2)$ values. In many situations, there are no discernible differences when comparing Exp and E$_3$ service, other than a decrease in the optimal cost under E$_3$ service. However, when comparing either Exp or E$_3$ against H$_2$ service, it is common to find different optimal $(k_1, k_2)$ pairs and we always observe an increase in the optimal cost.

Although there exists some evidence to suggest that the optimal $(k_1, k_2)$ values are, more or less, insensitive to the second moment of $Ser_i$ in our model (and this is consistent with the remarks in Borst et al. [17]), we did capture varying results by inflating the differences in variance between the two service time distributions to a large enough degree. One may be inclined to attribute the presence of these observed changes in our optimal results to only the occasional high blocking probability, rather than the service time distribution, but we must emphasize that some of these variations were still present in instances with negligible blocking probabilities (e.g., when $\alpha_1 = \alpha_2 = 0.25$ and H$_2$ service times are used for both

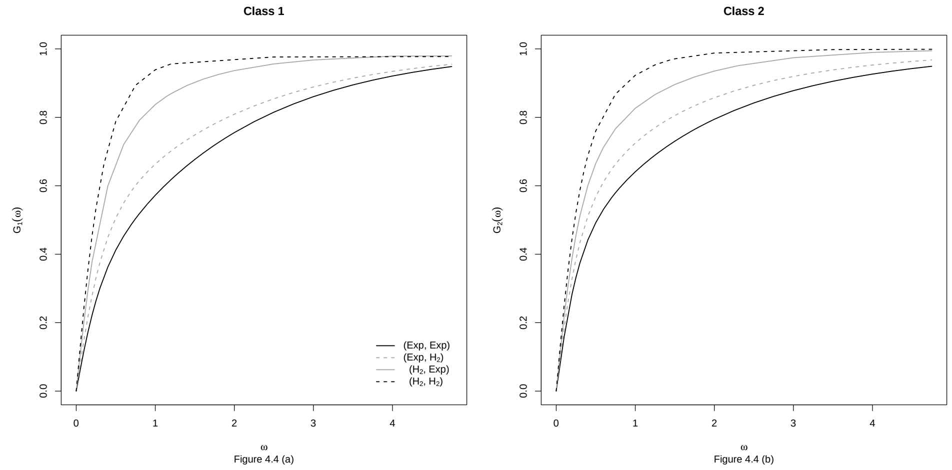

Figure 4.4: Plots of $G_i(\omega)$ versus $\omega$ for both classes under Case 2 with $\alpha_1 = 0.025$, $\alpha_2 = 0.25$, either Exp or $H_2$ service times, and optimal $(k_1, k_2)$ from Table 4.2.

classes). So, while blocking probabilities can contribute to the variability in the optimal $(k_1, k_2)$ combinations, we cannot conclude that the selection is completely insensitive as differences may exist even when variances are similar (e.g., (Exp, Exp) versus ($E_3$, Exp) services in Table 4.1). Furthermore, based on the form of Equation (4.16), another conclusion may be made. By selecting a larger arrival rate for a class, the expected time waiting in the system will increase, as well as the probability of reneging, while simultaneously raising that class's weight in the cost function. This will heighten the system's sensitivities to the service time distribution of that class, and can lead to more variation in the selection of optimal $(k_1, k_2)$ when comparing combinations of service time distribution for that class with smaller differences in variance.

### 4.5.2 Examining Actual Waiting Times and the Reduction Effect

Figures 4.3 and 4.4 present plots of $G_i(\omega)$, the distribution function of the actual waiting time random variable $W_i^*$, as defined in Section 4.4. These functions were evaluated via Equation (4.11) for both classes under a particular pair of reneging rates (namely, $\alpha_1 = 0.025$ and $\alpha_2 = 0.25$) and four combinations of Exp and $H_2$ service, with Cases 1 and 2 presented in Figures 4.3 and 4.4, respectively. For each combination of service time distribution, the optimal values of $k_1$ and $k_2$ were selected for use from Tables 4.1 and 4.2. It is interesting to note that $H_2$ service in these cases typically yielded shorter actual waiting times than Exp service. This is due to the fact that the actual waiting time distribution is conditional on the customer reaching service before reneging. In order to have the same expected value as Exp service (while inflating the variance), the selected $H_2$ service is constructed as a mixture of two exponential distributions, one with a higher rate and a great likelihood of occurrence (namely, 99.9%) and the other with a very low rate and a rare chance of occurrence (namely, 0.1%). The conditional nature of $W_i^*$ results in an exponentially distributed upper bound on the total time for the preceding customers' service and reneging times, implying that if the reneging rate of the target customer is high enough (so that the bound on the total time to reach service is short enough), we realistically can only observe services which follow the more common higher rate. This essentially reduces $H_2$ service to an exponential distribution with faster service times (and hence shorter actual waiting times).

Overall, we observe that the mean actual waiting times in Case 1 are primarily dependent on the class-1 service time distribution due to its larger mean service time (i.e., $\mu_1 = 0.9$ versus $\mu_2 = 0.1$). Combined with the fact that a high reneging rate for a particular class reduces the influence of that class's service time distribution, we witness the rather extreme situation seen in Figure 4.3, where the distribution is almost entirely dependent on class 1 (since $\alpha_1 = 0.025$ and $\alpha_2 = 0.25$). While the assumption of equal mean service times helped balance the dependence between classes in Case 2, the fact that the class-1 arrival rate is twice that of class 2 still resulted in a larger influence from class 1, as seen in Figure 4.4.

In order to better understand the behaviour of the actual waiting time distribution, we also calculated $\mathrm{E}[W_i^*]$ via Equation (4.14) for a variety of cases. In Figures 4.5 and 4.6, the expected actual waiting times for both classes are plotted against $k_1 = 1, 2, \ldots, 11$ (letting $k_2 = 12 - k_1$) for Case 1 and Case 2, respectively. Combinations of low (0.025) and high (0.25) reneging rates are considered, as well as combinations of Exp and $\mathrm{H}_2$ service.

In Figure 4.5, since $\mu_1 = 0.9 = 9\mu_2$, changing the distribution of class-1 services from Exp to $\mathrm{H}_2$ has the largest impact for the mean actual waiting times of either class. For class 1, changing the class-1 service distribution while keeping the class-2 service distribution the same causes a decrease in expected actual waiting time, as anticipated, given the previous discussion concerning the CDFs. This is observed for class 2 as well for larger $k_1$'s when $\alpha_2 = 0.025$, i.e., when the bound is weaker and it is possible that more class-1 services must be completed before the server can visit the class-2 queue, or for all $k_1$ when $\alpha_2 = 0.25$. When $k_1$ is small and the probabilistic bound is weak, there is less of a 'reducing' effect felt by the class-1 $\mathrm{H}_2$ services, and so rather than acting as faster exponential distributions, their increased service variance causes increases in the class-2 actual waiting times (a similar situation is present for small $k_2$ in the bottom right of Figures 4.5 (a) and (c), where class-2 $\mathrm{H}_2$ service results in slightly larger values). The effect's dependency on $k_1$ was not present for class 1 as a target class-1 customer must wait for all customers queued ahead of them to leave the system prior to reaching the server, themselves. Similarly, when $\alpha_i = 0.025$, changing class-2 services from Exp to $\mathrm{H}_2$ increases $\mathrm{E}[W_i^*]$, $i = 1, 2$. In this case, the smaller expected service times prevent the reducing effect, rather than a small $k_1$. When $\alpha_i = 0.25$, the effect is present again and switching to class-2 $\mathrm{H}_2$ service results in smaller expected actual waiting times.

It can also be noted that for both classes and either small or large reneging rates, $\mathrm{E}[W_i^*]$ is generally more sensitive to changes in $k_1$ when $Ser_1$ is Exp, rather than $\mathrm{H}_2$ (where the lines are generally flat for middle $k_1$ values). This may be another result of the $\mathrm{H}_2$ distributions acting as faster exponential distributions (also having smaller variances as a consequence), and hence the duration of a visit by the server to the class-1 queue will have smaller means and variances. As the server can empty the queue possessing larger jobs faster and more consistently, this will reduce the duration of said visits, simultaneously reducing the expected number of class-2 customers that will arrive between visits to their queue. This will lower the expected number of customers belonging to either class in the queue. If these lower queue lengths reduce occurrences of the server hitting their $k_i$ limit and having to switch prior to emptying a queue, then this will greatly lower the sensitivity of expected actual waiting times on a change of $k_1$, other than when a class's $k_i$ is very small (e.g., changed from 1 to 2), where the impact of this change is still likely to be felt.

In Figure 4.6, we now consider Case 2 where $\mu_1 = \mu_2 = 1$, meaning that a change in either class' service distribution will be significant. Since $\lambda_1 = 0.5 = 2\lambda_2$, class-1 will get on average twice as many customers, and hence the impact of the reducing effect will be felt by both classes more when applied to the service times of class-1 customers. Unlike before, changing class-2 services from Exp to $H_2$ will result in smaller expected actual waiting times for class 1 when $\alpha_1 = 0.025$ (since $\mu_2$ is much larger). The interaction of small $\alpha_2$ ($\alpha_1$) and small $k_1$ ($k_2$) where $H_2$ service is worse than Exp, as well as decreased sensitivity to $k_1$ under $H_2$ service, are similarly both present in this figure. Of course, class-2 $H_2$ service has more of an impact on reducing sensitivity than it did in Figure 4.5, albeit still smaller than that of class 1's service distribution.

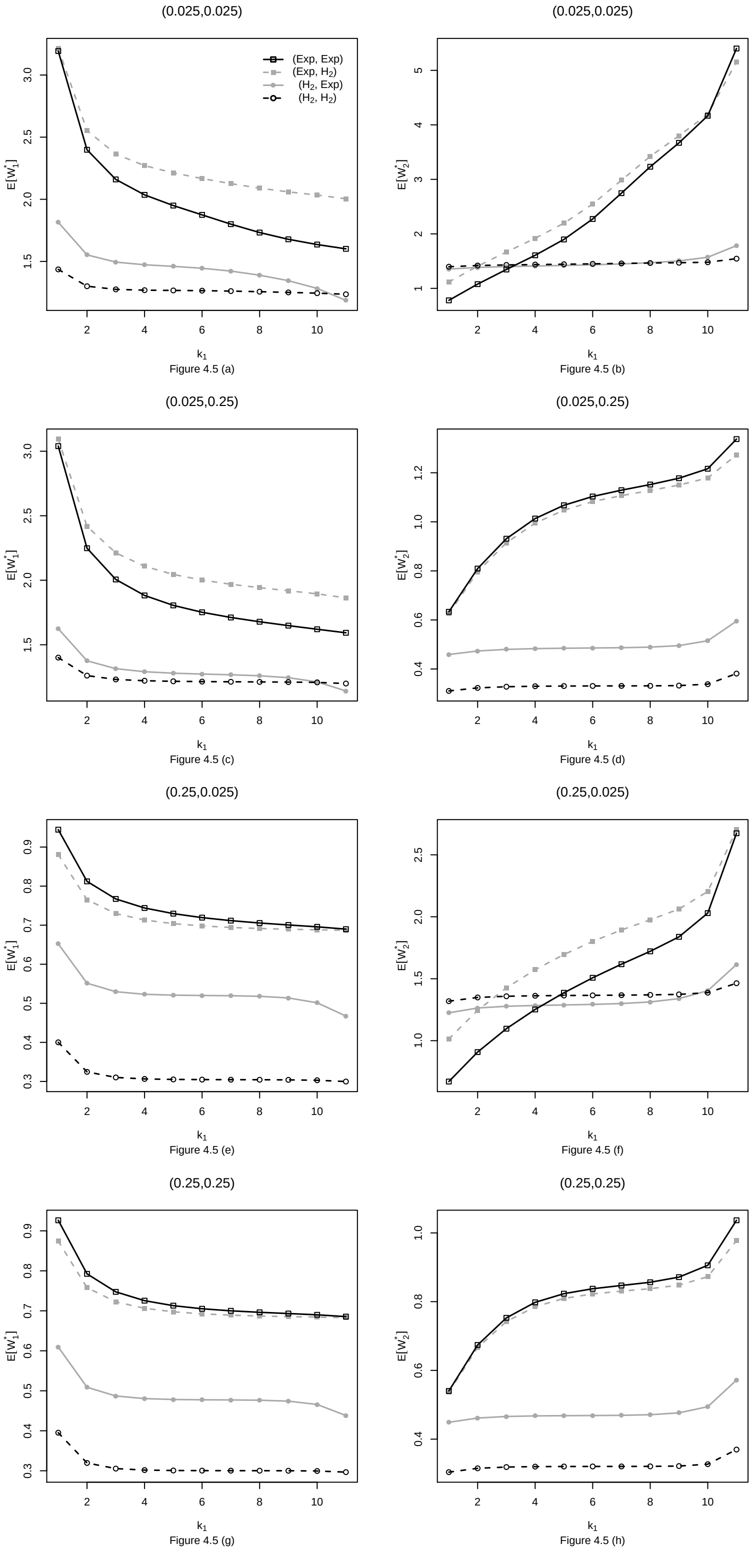

Figure 4.5 (a)

Figure 4.5 (b)

Figure 4.5 (c)

Figure 4.5 (d)

Figure 4.5 (e)

Figure 4.5 (f)

Figure 4.5 (g)

Figure 4.5 (h)

Figure 4.5: Plots of E$[W_i^*]$ versus $k_1$ for both classes and various $(\alpha_1, \alpha_2)$ under Case 1 and either Exp or H$_2$ service times.

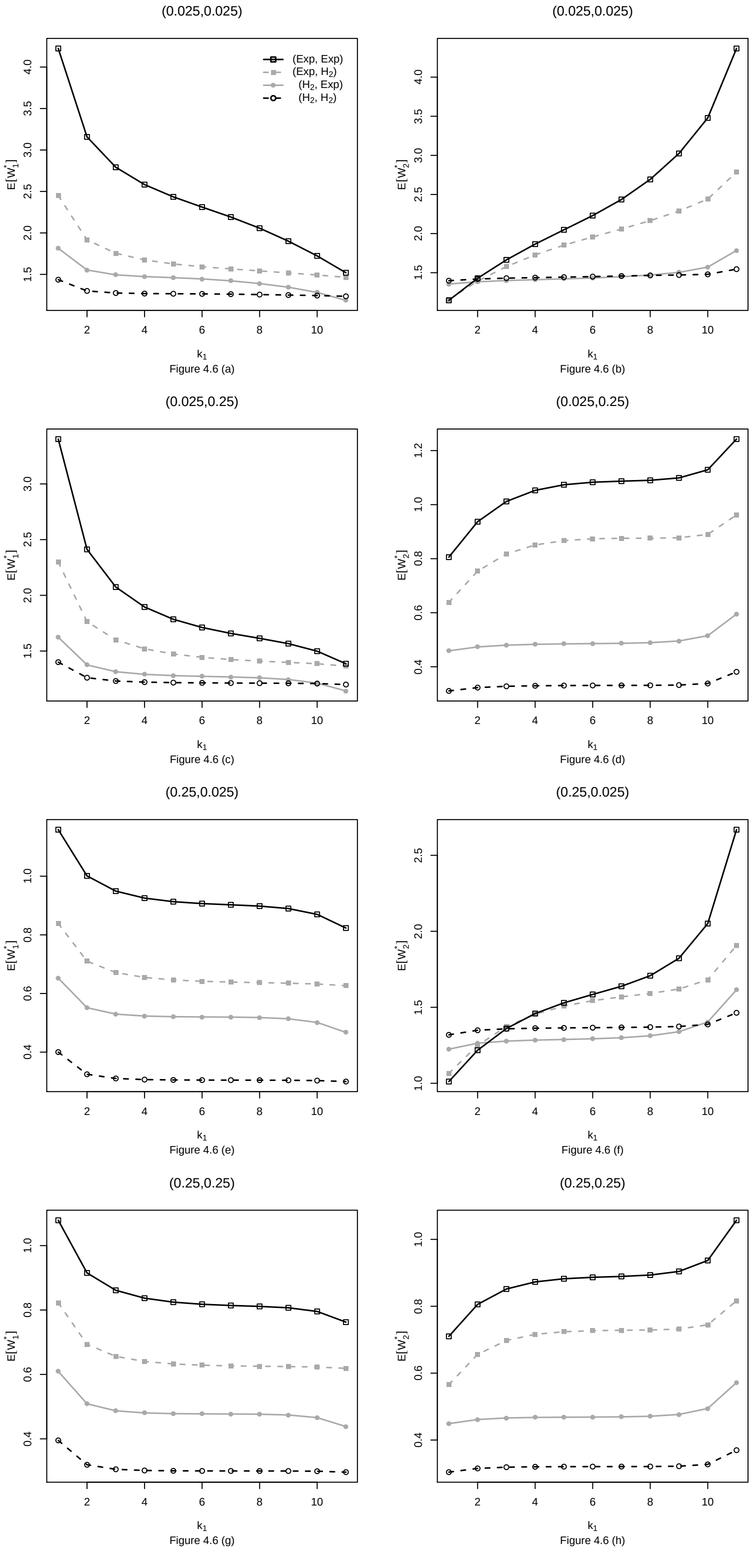

Figure 4.6 (a)

Figure 4.6 (b)

Figure 4.6 (c)

Figure 4.6 (d)

Figure 4.6 (e)

Figure 4.6 (f)

Figure 4.6 (g)

Figure 4.6 (h)

Figure 4.6: Plots of $\mathrm{E}[W_i^*]$ versus $k_1$ for both classes and various $(\alpha_1, \alpha_2)$ under Case 2 and either Exp or $\mathrm{H}_2$ service times.

# Chapter 5

# The Unobserved Waiting Customer Approximation

## 5.1 Discussion of Literature and Introduction to UWC

As we have seen through our preceding analyses, MAM allows for many flexibilities when modelling a queueing system or network. However, it does come with its share of limitations and restrictions. Within this chapter, we introduce the Unobserved Waiting Customer approximation which aims to improve the performance of MAM in a situation where it may struggle. Specifically, it aims to reduce the natural biases incurred from the required use of state truncation on a system that should in reality have infinite buffers. In the following sections, let IB denote the true *infinite buffer* model of interest, let FB denote the *finite buffer* model obtained through simple truncation, and let UWC denote the truncated model making use of the *Unobserved Waiting Customer* approximation.

For simple queues, such as level-independent QBDs, we may conduct an exact accurate analysis that considers all possible queue lengths (e.g., Section 1.2.4). However, to analyze more complicated queueing systems involving multiple queues and/or level-dependent QBD structures (e.g., due to reneging), we may be required to truncate the state space. If we say, remove all states beyond a threshold representing a queue length of $C$ customers, then it is typical to interpret the removal of these states as the enforcement of a finite buffer which is not present in the real world system we are trying to model. This inaccuracy will result in the steady-state probabilities of the removed states being redistributed proportionally to states belonging to lower queue lengths.

This was observed by Bright and Taylor [23] within their work considering how to numerically solve for the steady-state probabilities of a level-dependent QBD. They stated that element-wise, if the CTMC is positive recurrent, the steady-state probability for a state at a given truncation level is greater than or equal to the true value (which we may recover by letting the truncation level go to infinity). They discussed how to select the truncation level to ensure that the steady-state probability of the QBD being in a state at or above this level is negligible. One method is simply to iteratively increase the level until the sum of steady-state probabilities of all states at the truncation level is below a desired tolerance. This is similar to the approach used by Gertsbakh [37] when modelling a 2-queue system where an arriving customer joins the shortest queue. The level of their process was set to be the length of the shorter queue, while the longer queue is truncated to never be $n$ customers longer than the shorter queue. If the difference in queue length reached $n$, then it was assumed that a customer would immediately jockey to the shorter queue. In their numerical investigation, they selected a value of $n$ such that the steady-state probabilities for all states below the truncation level would change by less than $10^{-6}$ when further increasing the threshold level by 1.

Alternatively, Bright and Taylor [23] also investigated how to construct a dominating process which can be used to find an analytic upper bound on the tail probability, making use of normal birth-and-death process results. By ensuring that the upper bound of the tail probability is below a threshold, the true tail probability must also be acceptable. An example of applying this methodology of Bright and Taylor is the work of Krishnamoorthy et al. [57], in the context of a queue with self-promoting customers (which resulted in a level-dependent QBD). Rather than simply considering the tail probability, Kim and Kim [52] derived an upper bound for the truncation error in their $M/PH/1$ retrial queue with no waiting room, such that an arriving customer who did not find the server free immediately entered an orbit. Truncation error was defined as the sum of absolute-value differences in the steady-state probabilities for all states between their truncated model and the true IB model. They similarly used this upper bound to select a level at which to truncate their customer orbit such that the truncation error was below a specified tolerance.

Unfortunately, it is not always computationally feasible to use a truncation level of $C$ that is large enough to ensure that the tail probability or truncation error is below a small tolerance, especially if we are modelling a network of multiple queues all suffering from this issue simultaneously. It then benefits us to consider alternative modifications to a CTMC that give rise to results that outperform a simple FB model. For example, Diamond and Alfa [30] analyzed a retrial queue which tracked both the number of customers in the queue as well as in the retrial orbit. Similar to a 2-queue polling system, it is impossible to let both the queue and orbit have infinite buffers when using MAM. They elected to put a finite buffer on the queue, taking the number of customers in the orbit as the level of their QBD. They modified their CTMC so that after a certain level, if their queue is not full, then a customer will immediately enter it from the orbit. This approximation is fairly reasonable since as the level increases, the time between retrial attempts will go to zero. This leads to a level-independent QBD structure beyond this level, resulting in more accurate steady-state probabilities than simple truncation, and the level was selected at the point where the tail probability was below a given tolerance. Shin and Choo [86] used a similar approximation in that for part of their analysis of a $M/M/s$ retrial queue with customer balking and reneging, in order to enable the use of approximate analytical results, they assumed that the total effective reneging rate of customers in queue did not change beyond a certain level.

Differing from these adjustments, we propose the use of our UWC approximation to improve the overall numerical accuracy when approximating an infinite buffer system when we are unable to use a large enough $C$. Our ultimate goal is to reduce the negative bias in the expected value of queue lengths at steady state that results from state truncation. As we wish to apply this to polling models with potentially very large state spaces, we will do so without requiring the model to track additional states. Also, rather than altering the behaviour of customers in the system to create a level-independent structure, we will be approximating events that are unobservable by the model. Suppose that in a given queue we truncate at level $C$, such that we remove all states corresponding to queue lengths greater than $C$. Rather than assuming the presence of a finite buffer, we assume that customers may be present in positions $C + 1, C + 2, \ldots$, but are unobservable. If the observed portion of the queue is full, then following an observed customer departure, an unobserved waiting customer may immediately fill the available observed position.

The goal of the UWC approximation is to aggregate probability mass from the tail to the truncation level, resulting in steady-state probabilities at states below the buffer that are either unbiased or less biased than those in a standard FB model. While not designed for level-dependent QBDs, ETAQA (an acronym for "an Efficient Technique for the Analysis of QBD-processes by Aggregation") is an example of an aggregation method for level-independent QBDs. ETAQA was introduced by Ciardo and Smirni [29] for level-independent QBDs satisfying the restriction that all transitions which reduce the level of the QBD transition into the same sublevel, and was extended to $M/G/1$-type CTMCs by Riska and Smirni [81]. Specifically, their ETAQA method calculated steady-state probability vectors $\underline{\pi}_0$, $\underline{\pi}_1$, and $\underline{\pi}^* = \sum_{i=2}^{\infty} \underline{\pi}_i$, where $\underline{\pi}_0$ and $\underline{\pi}_1$ are unbiased and in the latter vector, all sublevels across higher levels are grouped into individual states (i.e., $\pi_j^* = \sum_{i=2}^{\infty} \pi_{i,j}$). Heindl [44] would later show that ETAQA for level-independent QBDs could actually use any level for state aggregation, not just level 2.

Differing from ETAQA, an advantage to UWC is that it may be used to improve accuracy in more general cases where it does not yield exact results. The ability to apply UWC to level-dependent QBDs is also of more use in general than being limited to level-independent QBDs (we note, however, that the goal of ETAQA is not to circumvent truncation limitations, but rather to provide a quicker alternative to solve for things such as linear combinations of queue length moments).

## 5.2 $M/M/1$ Queue

We begin by considering the classic $M/M/1$ queue we previously examined in Section 1.2.2. In this queue, customer arrivals are governed by a Poisson process with intensity $\lambda$ and customers are served individually by a single server according to independent and identically distributed (iid) service times with distribution $Ser \sim \text{Exp}(\mu)$. Let $\pi_i$ be the steady-state probability of observing $i$ customers in the IB model, $i \in \mathbb{N}$. The balance equations for the IB model may be expressed as

$$\lambda \pi_i = \mu \pi_{i+1}, \ i \in \mathbb{N},$$

which in combination with the normalization condition, $1 = \sum_{i=0}^{\infty} \pi_i$, we have seen satisfy

$$\pi_i = \rho^i (1-\rho), \ i \in \mathbb{N},$$

provided that $\rho = \lambda/\mu < 1$. Letting $X^{\text{IB}}$ denote the IB model queue length at steady state, it follows that

$$\text{E}[X^{\text{IB}}] = \frac{\rho}{1-\rho}.$$

Considering now the simple truncation case, we let $\pi_i^{\text{FB}}$ be the steady-state probability of observing $i$ customers in the FB model, $i = 0, 1, \ldots, C$. The corresponding modified balance equations are

$$\lambda \pi_i^{\text{FB}} = \mu \pi_{i+1}^{\text{FB}}, \ i = 0, 1, \ldots, C-1,$$

which in combination with the normalization condition, $1 = \sum_{i=0}^{C} \pi_i^{\text{FB}}$, results in

$$\pi_i^{\text{FB}} = \frac{\rho^i (1-\rho)}{1-\rho^{C+1}} = \frac{\pi_i}{1-\rho^{C+1}}, \ i = 0, 1, \ldots, C,$$

which are equal to the first $C+1$ steady-state probabilities from the IB model, re-normalized. That is, all probability mass above state $C$ is proportionally redistributed across the lower states. If we similarly let $X^{\mathrm{FB}}$ denote the FB model queue length at steady state, then we can show that

$$\mathrm{E}[X^{\mathrm{FB}}] = \frac{\rho}{1-\rho} - \frac{(C+1)\rho^{C+1}}{1-\rho^{C+1}},$$

which clearly demonstrates a negatively biased mean queue length, relative to the IB model.

We will now introduce our UWC approximation to adjust the system so that this negative bias will be reduced. In the FB model, the implication of the buffer is that a customer who observes a queue length of $C$ at their arrival instant will be blocked and be lost. Instead, we suppose that these customers can still wait in the queue, however they are unobserved by the system. As they are not tracked, we must instead approximate their presence. We do so by introducing a probability $p_C^*$ of there being one or more unobserved customers present in the queue at an observed customer's departure epoch. In this way, with probability $p_C^*$, there will be a customer present who will immediately fill the vacant observable position within the queue following the departure, and hence the observed queue length does not decrement from our perspective.

As we are not introducing any new states and must preserve the Markov property within our analytical framework, this probability is a constant and cannot depend on how many times unobserved customers have entered into the observable portion of the queue in this way. Therefore, the distribution of how many observed customer departures are required to decrement the queue length below the buffer is geometric with success probability $1-p_C^*$. It follows that the effective amount of time spent in state $C$ in the UWC model has an $\mathrm{Exp}((1-p_C^*)\mu)$ distribution. Letting $\pi_i^{\mathrm{UWC}}$ be the steady-state probability of observing $i$ customers in the UWC model, $i=0,1,\ldots,C$, the balance equations are

$$\lambda\pi_i^{\mathrm{UWC}} = \mu\pi_{i+1}^{\mathrm{UWC}},\ i=0,1,\ldots,C-2, \tag{5.1}$$

$$\lambda\pi_{C-1}^{\mathrm{UWC}} = (1-p_C^*)\mu\pi_C^{\mathrm{UWC}}. \tag{5.2}$$

We must now determine an appropriate choice for $p_C^*$. We elect to choose a $p_C^*$ which requires the same expected number of observed customer departures (in this case, solely from service completions) in the UWC model to transition from state $C$ to $C-1$ (namely, $(1-p_C^*)^{-1}$), as in the IB model between a visitation instant to state $C$ until it returns to state $C-1$ for the first time (we will henceforth refer to this type of time interval as a level-$C$ busy period). As the $M/M/1$ queue has level-independent service rates, the distribution of a level-$i$ busy period is independent of $i$. That is, it has an identical distribution to a standard busy period.

From Equation (1.9), we know that the expected value of a busy period in an $M/M/1$ queue is $\mathrm{E}[BP]=(\mu-\lambda)^{-1}$. The entire busy period consists of some random number of sequential service times, with no server idling (and no setup time prior to the first service). Therefore, the expected number of service completions during a busy period is simply $\mathrm{E}[BP]/\mathrm{E}[Ser]$, and so we set

$$\frac{1}{1-p_C^*} = \frac{\mathrm{E}[BP]}{\mathrm{E}[Ser]} = \frac{\mu}{\mu-\lambda} = \frac{1}{1-\rho},$$

implying that $p_C^*=\rho$. Substituting this value into Equations (5.1) and (5.2) and solving with the normalization condition $1 = \sum_{i=0}^{C} \pi_i^{\mathrm{UWC}}$, we find

$$\pi_0^{\mathrm{UWC}} = \left( \sum_{i=0}^{C-1} \rho^i + \frac{\rho^C}{1-\rho} \right)^{-1} = \left( \frac{1-\rho^C}{1-\rho} + \frac{\rho^C}{1-\rho} \right)^{-1} = 1 - \rho = \pi_0,$$

$$\pi_i^{\mathrm{UWC}} = \rho^i \pi_0^{\mathrm{UWC}} = \rho^i(1-\rho) = \pi_i, \ i = 1, 2, \ldots, C-1,$$

and

$$\pi_C^{\mathrm{UWC}} = \frac{\rho^C}{1-\rho} \pi_0^{\mathrm{UWC}} = \rho^C = \sum_{i=C}^{\infty} \rho^i (1-\rho) = \sum_{i=C}^{\infty} \pi_i.$$

In contrast to the FB model, the UWC model has unbiased steady-state probabilities for states $i = 0, 1, \ldots, C-1$, while allocating all excess probability mass for states at or above the buffer $C$ into $\pi_C^{\mathrm{UWC}}$. Letting $X^{\mathrm{UWC}}$ denote the UWC model queue length at steady state, we can confirm that

$$\mathrm{E}[X^{\mathrm{UWC}}] = \frac{\rho}{1-\rho} - \frac{\rho^{C+1}}{1-\rho},$$

which is strictly between the mean queue lengths of the FB and IB models, and so this achieves our goal of reducing the negative bias inherent to truncation.

## 5.3 $M/M/1 + M$ Queue

We next consider the $M/M/1$ queue as outlined in Section 5.2, while further supposing that any customers who are not actively being served are at risk of reneging from the queue due to their own iid $\mathrm{Exp}(\alpha)$ impatience timers. The balance equations for the IB model of this queue are

$$\lambda \pi_i = (\mu + i\alpha)\pi_{i+1}, \ i \in \mathbb{N}.$$

Under the normalization condition $1 = \sum_{i=0}^{\infty} \pi_i$, we obtain the solution

$$\pi_i = \frac{\lambda^i \left( \prod_{j=0}^{i-1} (\mu + j\alpha) \right)^{-1}}{1 + \sum_{k=1}^{\infty} \lambda^k \left( \prod_{j=0}^{k-1} (\mu + j\alpha) \right)^{-1}}, \ i \in \mathbb{N}, \tag{5.3}$$

where we use the convention $\prod_{j=0}^{0-1} (\mu + j\alpha) = 1$.

Unlike in the $M/M/1$ queue, the choice of our UWC probability will now depend on $C$. This is true under the assumption that $\alpha > 0$, since a level-$C$ busy period in either model will be shorter due to the larger total effective rate of customer departures as a result of reneging from queue positions 2 through $C$. The modified balance equations for the UWC model are

$$\begin{aligned} \lambda \pi_i^{\mathrm{UWC}} &= (\mu + i\alpha)\pi_{i+1}^{\mathrm{UWC}}, \ i = 0, 1, \ldots, C-2, \\ \lambda \pi_{C-1}^{\mathrm{UWC}} &= (1 - p_C^*)(\mu + (C-1)\alpha)\pi_C^{\mathrm{UWC}}, \end{aligned}$$

which, when solved along with the normalization condition $1 = \sum_{i=0}^{C} \pi_i^{\mathrm{UWC}}$, yields the solution

$$\pi_0^{\mathrm{UWC}} = \left( 1 + \sum_{k=1}^{C-1} \frac{\lambda^i}{\prod_{j=0}^{i-1} (\mu + j\alpha)} + \frac{1}{1 - p_C^*} \cdot \frac{\lambda^C}{\prod_{j=0}^{C-1} (\mu + j\alpha)} \right)^{-1}, \tag{5.4}$$

$$\pi_i^{\text{UWC}} = \frac{\lambda^i}{\prod_{j=0}^{i-1}(\mu + j\alpha)}\pi_0^{\text{UWC}} = \frac{\frac{\lambda^i}{\prod_{j=0}^{i-1}(\mu+j\alpha)}}{1 + \sum_{k=1}^{C-1}\frac{\lambda^i}{\prod_{j=0}^{i-1}(\mu+j\alpha)} + \frac{1}{1-p_C^*}\cdot\frac{\lambda^C}{\prod_{j=0}^{C-1}(\mu+j\alpha)}}, \tag{5.5}$$

for $i = 1, 2, \ldots, C-1$, and

$$\pi_C^{\text{UWC}} = \frac{1}{1-p_C^*}\cdot\frac{\lambda^C}{\prod_{j=0}^{C-1}(\mu+j\alpha)}\pi_0^{\text{UWC}} = \frac{\frac{1}{1-p_C^*}\cdot\frac{\lambda^C}{\prod_{j=0}^{C-1}(\mu+j\alpha)}}{1+\sum_{k=1}^{C-1}\frac{\lambda^i}{\prod_{j=0}^{i-1}(\mu+j\alpha)}+\frac{1}{1-p_C^*}\cdot\frac{\lambda^C}{\prod_{j=0}^{C-1}(\mu+j\alpha)}}. \tag{5.6}$$

We can obtain $\pi_i^{\text{FB}}$, $i = 0, 1, \ldots, C$, by simply setting $p_C^* = 0$ in Equations (5.4)-(5.6), resulting in

$$\pi_i^{\text{FB}} = \pi_i \cdot \frac{1 + \sum\limits_{k=1}^{\infty}\lambda^k\left(\prod\limits_{j=0}^{k-1}(\mu + j\alpha)\right)^{-1}}{1 + \sum\limits_{k=1}^{C}\lambda^k\left(\prod\limits_{j=0}^{k-1}(\mu + j\alpha)\right)^{-1}} > \pi_i,\ i = 0, 1, \ldots, C.$$

We will now select $p_C^*$ in a similar manner as before, equating the expected number of observed customer departures (either from service completions or reneging from the first $C$ queue positions) during a level-$C$ busy period in the UWC and IB models. The distribution of a level-$C$ busy period in the IB model will be identically distributed as a standard busy period of a $M/M/1+M$ queue with service rate $\mu + (C-1)\alpha$ and individual customer reneging rate $\alpha$. That is, we group the reneging of all customers at or before the truncation level with the service rate of the leading customer to get an effective overall service rate, since we only care about departures and do not distinguish between ways that customers may leave the system. Let this effective service time be represented by the random variable $Ser_C \sim \text{Exp}(\mu + (C-1)\alpha)$.

In order to solve for the mean busy period of a $M/M/1+M$ queueing system, we make use of Equation (1.10) from Remark 1.2, which allows us to express it in terms of the server's idle probability. Letting $i = 0$ and replacing $\mu$ by $\mu + (C-1)\alpha$ in Equation (5.3), we apply Equation (1.10) to ultimately obtain

$$\text{E}[BP_C] = \sum_{k=1}^{\infty}\lambda^{k-1}\left(\prod_{j=0}^{k-1}(\mu + (C-1+j)\alpha)\right)^{-1},$$

where we let $BP_C$ denote the IB model level-$C$ busy period. Finally, we set

$$\frac{1}{1-p_C^*} = \frac{\text{E}[BP_C]}{\text{E}[Ser_C]} = \sum_{k=1}^{\infty}\frac{\lambda^{k-1}(\mu + (C-1)\alpha)}{\prod_{j=0}^{k-1}(\mu + (C-1+j)\alpha)}, \tag{5.7}$$

implying that

$$p_C^* = 1 - \left(\sum_{k=1}^{\infty}\frac{\lambda^{k-1}(\mu + (C-1)\alpha)}{\prod_{j=0}^{k-1}(\mu + (C-1+j)\alpha)}\right)^{-1}. \tag{5.8}$$

Observe that

$$\begin{aligned}\frac{1}{1-p_C^*}\cdot\frac{\lambda^C}{\prod_{j=0}^{C-1}(\mu+j\alpha)} &= \left(\sum_{k=1}^{\infty}\frac{\lambda^{k-1}(\mu+(C-1)\alpha)}{\prod_{j=0}^{k-1}(\mu+(C-1+j)\alpha)}\right)\frac{\lambda^C}{\prod_{j=0}^{C-1}(\mu+j\alpha)}\\ &= \sum_{i=C}^{\infty}\frac{\lambda^i}{\prod_{j=0}^{i-1}(\mu+j\alpha)}.\end{aligned}$$

If we substitute Equation (5.7) into Equations (5.4)-(5.6), we can recover $\pi_i^{\text{UWC}} = \pi_i$, $i = 0, 1, \ldots, C-1$, and

$$\pi_C^{\text{UWC}} = \sum_{i=C}^{\infty}\frac{\lambda^i\left(\prod\limits_{j=0}^{i-1}(\mu+j\alpha)\right)^{-1}}{1+\sum\limits_{k=1}^{\infty}\lambda^k\left(\prod\limits_{j=0}^{k-1}(\mu+j\alpha)\right)^{-1}} = \sum_{i=C}^{\infty}\pi_i.$$

As in the $M/M/1$ system, the UWC model accurately calculates the steady-state probabilities below the truncation level with no bias, while collecting all excess probability mass into $\pi_C^{\text{UWC}}$. Note that if we set $\alpha = 0$, Equation (5.8) simplifies to give

$$p_C^* = 1-\left(\sum_{k=1}^{\infty}\frac{\lambda^{k-1}}{\mu^{k-1}}\right)^{-1} = 1-\left(\frac{1}{1-\rho}\right)^{-1} = \rho,$$

recovering $p_C^*$ from the the $M/M/1$ model, as required.

## 5.4 $M/M/\infty$ Queue

By letting $\alpha = \mu$ in our results from Section 5.3, we immediately recover the analysis for a $M/M/\infty$ queue where every customer immediately begins an iid Exp($\mu$) service time upon entering the system. In summary,

$$\pi_i = \frac{\rho^i}{i!}e^{-\rho},\ i\in\mathbb{N},$$

$$\pi_i^{\text{FB}} = \pi_i\cdot\frac{e^{\rho}}{\sum_{k=0}^{C}\rho^k/k!} > \pi_i,\ i = 0,1,\ldots,C,$$

$$p_C^* = 1-\frac{\rho^C}{C!}\left(e^{\rho}-\sum_{k=0}^{C-1}\frac{\rho^k}{k!}\right)^{-1}, \tag{5.9}$$

and it remains that $\pi_i^{\text{UWC}} = \pi_i$, $i = 0, 1, \ldots, C-1$, and $\pi_C^{\text{UWC}} = \sum_{i=C}^{\infty}\pi_i$.

## 5.5 $M/PH/1$ Queue

We now consider an analogous model to the $M/M/1$ queue, however we generalize the customer service time distribution from $Ser \sim \text{Exp}(\mu)$ to $Ser \sim \text{PH}_b(\underline{\beta}, B)$. That is, service times are iid continuous phase-type random variables of order $b$, and we assume that $\underline{\beta}\underline{e}' = \sum_{i=1}^{b} \beta_i = 1$, indicating that service times must be strictly positive in duration. We assume that $\lambda \mathrm{E}[Ser] < 1$ to guarantee stability in the model.

We have previously considered the analysis of a $M/PH/1$ queue as a level-independent QBD in Section 1.2.4 as a way to introduce the matrix geometric solution. We now demonstrate the analytic solution of this queue before approaching the application of UWC. The $M/PH/1$ queue is modelled using a CTMC denoted by $\{(X(t), Y(t)), t \geq 0\}$, where $X(t)$ is the number of customers in the system and $Y(t)$ is the current service phase at time $t$, which has possible values depending on $X(t)$:

$$Y(t) \in \Omega_Y(X(t)) = \begin{cases} \{0\} & \text{, if } X(t) = 0, \\ \{1, 2, \ldots, b\} & \text{, if } X(t) \in \mathbb{Z}^+. \end{cases}$$

Letting $X(t)$ denote the level of the process and allowing $Q_{i,j}$ to contain the rates corresponding to transitions where $X(t)$ would change from $i$ to $j$, the infinitesimal generator matrix for this queue takes on a QBD form

$$Q = \begin{array}{c|cccccccc|} & 0 & 1 & 2 & \cdots & C-1 & C & C+1 & \cdots \\ \hline 0 & Q_{0,0} & Q_{0,1} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots \\ 1 & Q_{1,0} & Q_{1,1} & Q_{1,2} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots \\ 2 & \mathbf{0} & Q_{2,1} & Q_{2,2} & \ddots & \ddots & \mathbf{0} & \mathbf{0} & \cdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots & \cdots \\ C-1 & \mathbf{0} & \mathbf{0} & \ddots & \ddots & Q_{C-1,C-1} & Q_{C-1,C} & \mathbf{0} & \cdots \\ C & \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & Q_{C,C-1} & Q_{C,C} & Q_{C,C+1} & \ddots \\ C+1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & Q_{C+1,C} & Q_{C+1,C+1} & \ddots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \ddots \end{array}, \tag{5.10}$$

Defining $I$ as an appropriately dimensioned identity matrix and $\underline{B}_0' = -B\underline{e}'$ as the column vector of absorption rates for rate matrix $B$, the generator blocks for the $M/PH/1$ queue are

$$\begin{array}{rclrclrcl} & & & Q_{0,0} & = & -\lambda, & Q_{0,1} & = & \lambda\underline{\beta}, \\ Q_{1,0} & = & \underline{B}_0', & Q_{1,1} & = & B - \lambda I, & Q_{1,2} & = & \lambda I, \\ Q_{i,i-1} & = & \underline{B}_0'\underline{\beta}, & Q_{i,i} & = & B - \lambda I, & Q_{i,i+1} & = & \lambda I, \ i = 2, 3, \ldots. \end{array} \tag{5.11}$$

As $Q_{i,j}$, $j = i-1, i, i+1$, do not change with $i$, $i \geq 2$, this is a level-independent QBD. Letting $\pi_{i,j}$ be the steady-state probability of observing the CTMC in state $(i, j)$ and partitioning the steady-state distribution as $\underline{\pi} = (\underline{\pi}_0, \underline{\pi}_1, \underline{\pi}_2, \ldots)$, where $\underline{\pi}_0 = \pi_{0,0}$ and $\underline{\pi}_i = (\pi_{i,1}, \pi_{i,2}, \ldots, \pi_{i,b})$, $i \in \mathbb{Z}^+$, we obtain from Equations (5.10) and (5.11)

$$0 = \pi_{0,0}(-\lambda) + \underline{\pi}_1\underline{B}_0', \tag{5.12}$$

$$\underline{0} = \pi_{0,0}(\lambda\underline{\beta}) + \underline{\pi}_1(B - \lambda I) + \underline{\pi}_2\underline{B}_0'\underline{\beta}, \tag{5.13}$$

$$\underline{0} = \underline{\pi}_i(\lambda I) + \underline{\pi}_{i+1}(B - \lambda I) + \underline{\pi}_{i+2}\underline{B}_0'\underline{\beta}, \ i \in \mathbb{Z}^+. \tag{5.14}$$

From Equation (5.12), it immediately follows that

$$\lambda\pi_{0,0} = \underline{\pi}_1\underline{B}_0'. \tag{5.15}$$

After post-multiplying Equations (5.13) and (5.14) by $\underline{e}'$ and performing some elementary substitutions, we can similarly obtain

$$\lambda \underline{\pi}_i \underline{e}' = \underline{\pi}_{i+1} \underline{B}'_0, \ i \in \mathbb{Z}^+. \tag{5.16}$$

Substituting Equation (5.16) for $i = 1$ into Equation (5.13) and solving for $\underline{\pi}_1$, we find

$$\underline{\pi}_1 = \pi_{0,0} \underline{\beta} \lambda (\lambda I - \lambda \underline{e}' \underline{\beta} - B)^{-1}, \tag{5.17}$$

and from Equations (5.14), (5.16), and (5.17),

$$\underline{\pi}_i = \underline{\pi}_{i-1} \lambda (\lambda I - \lambda \underline{e}' \underline{\beta} - B)^{-1} = \pi_{0,0} \underline{\beta} \lambda^i (\lambda I - \lambda \underline{e}' \underline{\beta} - B)^{-i}, \ i \in \mathbb{Z}^+. \tag{5.18}$$

Letting $R = \lambda(\lambda I - \lambda \underline{e}' \underline{\beta} - B)^{-1}$, we obtain the matrix geometric solution $\underline{\pi}_i = \underline{\pi}_1 R^{i-1}$, $i \in \mathbb{Z}^+$. Finally, the normalization condition is

$$1 = \underline{\pi}\, \underline{e}' = \pi_{0,0} + \sum_{i=1}^{\infty} \underline{\pi}_i \underline{e}' = \pi_{0,0} + \underline{\pi}_1 (I - R)^{-1} \underline{e}' = \pi_{0,0} \left(1 + \underline{\beta} R (I - R)^{-1} \underline{e}'\right). \tag{5.19}$$

We can confirm that

$$R(I - R)^{-1} \underline{e}' = \lambda (1 - \lambda \mathrm{E}[Ser])^{-1} (-B^{-1} \underline{e}'), \tag{5.20}$$

where $\mathrm{E}[Ser] = -\underline{\beta} B^{-1} \underline{e}'$. Substituting Equation (5.20) into Equation (5.19) and solving for $\pi_{0,0}$, we obtain

$$\pi_{0,0} = \left(1 + \frac{\lambda \mathrm{E}[Ser]}{1 - \lambda \mathrm{E}[Ser]}\right)^{-1} = 1 - \lambda \mathrm{E}[Ser].$$

Therefore, by Equation (5.18), the remaining steady-state probabilities for the $M/PH/1$ IB model are

$$\underline{\pi}_i = (1 - \lambda \mathrm{E}[Ser]) \underline{\beta} \lambda^i (\lambda I - \lambda \underline{e}' \underline{\beta} - B)^{-i}, \ i \in \mathbb{Z}^+. \tag{5.21}$$

We now consider the UWC model, and confirm that we can recover unbiased steady-state probabilities for levels $0, 1, \ldots, C-1$. Define the partitioned row vector of steady-state probabilities for the truncated CTMC applying the UWC approximation by

$$\underline{\pi}^{\mathrm{UWC}} = (\pi_{0,0}^{\mathrm{UWC}}, \underline{\pi}_1^{\mathrm{UWC}}, \ldots, \underline{\pi}_C^{\mathrm{UWC}}).$$

The generator blocks $Q_{i,j}^{\mathrm{UWC}}$ are only adjusted for $i \geq C$, such that the blocks which do not contain only zeroes are:

$$\begin{array}{rclrclrcl} & & & Q_{0,0}^{\mathrm{UWC}} & = & -\lambda, & Q_{0,1}^{\mathrm{UWC}} & = & \lambda \underline{\beta}, \\ Q_{1,0}^{\mathrm{UWC}} & = & \underline{B}'_0, & Q_{1,1}^{\mathrm{UWC}} & = & B - \lambda I, & Q_{1,2}^{\mathrm{UWC}} & = & \lambda I, \\ Q_{i,i-1}^{\mathrm{UWC}} & = & \underline{B}'_0 \underline{\beta}, & Q_{i,i}^{\mathrm{UWC}} & = & B - \lambda I, & Q_{i,i+1}^{\mathrm{UWC}} & = & \lambda I, \ i = 2, 3, \ldots, C-1, \\ Q_{C,C-1}^{\mathrm{UWC}} & = & (1 - p_C^*) \underline{B}'_0 \underline{\beta}, & Q_{C,C}^{\mathrm{UWC}} & = & B + p_C^* \underline{B}'_0 \underline{\beta}, & & & \end{array} \tag{5.22}$$

Here, we no longer observe arrivals at level $C$, and with probability $p_C^*$, there is at least one unobserved customer ready to enter the observed states at the time of a service completion, so we have $Q_{C,C}^{\mathrm{UWC}} = Q_{C,C} + \lambda I + p_C^* Q_{C,C-1}$, while we also set $Q_{C,C-1}^{\mathrm{UWC}} = (1 - p_C^*) Q_{C,C-1}$ and $Q_{C,C+1} = \mathbf{0}$.

From Equations (5.10) and (5.22), we have

$$\begin{aligned}
&0 = \pi_{0,0}^{\mathrm{UWC}}(-\lambda) + \underline{\pi}_1^{\mathrm{UWC}}\underline{B}_0', \\
&\underline{0} = \pi_{0,0}^{\mathrm{UWC}}(\lambda\underline{\beta}) + \underline{\pi}_1^{\mathrm{UWC}}(B - \lambda I) + \underline{\pi}_2^{\mathrm{UWC}}\underline{B}_0'\underline{\beta}, \\
&\underline{0} = \underline{\pi}_i^{\mathrm{UWC}}(\lambda I) + \underline{\pi}_{i+1}^{\mathrm{UWC}}(B - \lambda I) + \underline{\pi}_{i+2}^{\mathrm{UWC}}\underline{B}_0'\underline{\beta},\ i = 1, 2, \ldots, C-3, \\
&\underline{0} = \underline{\pi}_{C-2}^{\mathrm{UWC}}(\lambda I) + \underline{\pi}_{C-1}^{\mathrm{UWC}}(B - \lambda I) + \underline{\pi}_C^{\mathrm{UWC}}(1 - p_C^*)\underline{B}_0'\underline{\beta}, \qquad (5.23) \\
&\underline{0} = \underline{\pi}_{C-1}^{\mathrm{UWC}}(\lambda I) + \underline{\pi}_C^{\mathrm{UWC}}(B + p_C^*\underline{B}_0'\underline{\beta}),
\end{aligned}$$

from which we can obtain

$$\begin{aligned}
\lambda\pi_{0,0}^{\mathrm{UWC}} &= \underline{\pi}_1^{\mathrm{UWC}}\underline{B}_0', && (5.24) \\
\lambda\underline{\pi}_i^{\mathrm{UWC}}\underline{e}' &= \underline{\pi}_{i+1}\underline{B}_0',\ i = 1, 2, \ldots, C-2, && (5.25) \\
\lambda\underline{\pi}_{C-1}^{\mathrm{UWC}}\underline{e}' &= \underline{\pi}_C(1 - p_C^*)\underline{B}_0'.
\end{aligned}$$

Since Equations (5.24) and (5.25) have the same form as Equations (5.15) and (5.16), we similarly find that

$$\underline{\pi}_i^{\mathrm{UWC}} = \pi_{0,0}^{\mathrm{UWC}}\underline{\beta}\lambda^i(\lambda I - \lambda\underline{e}'\underline{\beta} - B)^{-i},\ i = 1, 2, \ldots, C-1. \tag{5.26}$$

However, substituting Equation (5.25) for $i = C-2$ into Equation (5.23) and solving for $\underline{\pi}_C^{\mathrm{UWC}}$, we have

$$\begin{aligned}
\underline{\pi}_C^{\mathrm{UWC}} &= \underline{\pi}_{C-1}^{\mathrm{UWC}}\lambda(-(B + p_C^*\underline{B}_0'\underline{\beta})^{-1}) \\
&= \pi_{0,0}^{\mathrm{UWC}}\underline{\beta}\lambda^C(\lambda I - \lambda\underline{e}'\underline{\beta} - B)^{-(C-1)}(-(B + p_C^*\underline{B}_0'\underline{\beta})^{-1}).
\end{aligned}$$

The normalization condition for the UWC model becomes

$$\begin{aligned}
1 = \underline{\pi}^{UWC}\underline{e}' &= \pi_{0,0}^{UWC} + \sum_{i=1}^{C}\underline{\pi}_i^{UWC}\underline{e}' \\
&= \pi_{0,0}^{UWC}\left(1 + \sum_{i=1}^{C-1}\underline{\beta}R^i\underline{e}' + \underline{\beta}R^{C-1}\lambda(-(B + p_C^*\underline{B}_0'\underline{\beta})^{-1}\underline{e}')\right). \qquad (5.27)
\end{aligned}$$

Note that we can alternately express Equation (5.19) as

$$1 = \pi_{0,0}\left(1 + \sum_{i=1}^{C-1}\underline{\beta}R^i\underline{e}' + \sum_{i=C}^{\infty}\underline{\beta}R^i\underline{e}'\right) = \pi_{0,0}\left(1 + \sum_{i=1}^{C-1}\underline{\beta}R^i\underline{e}' + \underline{\beta}R^C(I - R)^{-1}\underline{e}'\right),$$

so if $p_C^*$ satisfies

$$\underline{\beta}R^{C-1}\lambda(-(B + p_C^*\underline{B}_0'\underline{\beta})^{-1})\underline{e}' = \underline{\beta}R^C(I - R)^{-1}\underline{e}', \tag{5.28}$$

then $\pi_{0,0}^{\mathrm{UWC}} = \pi_{0,0}$, and by Equation (5.26), $\underline{\pi}_i^{\mathrm{UWC}} = \underline{\pi}_i$, $i = 1, 2, \ldots, C-1$.

We must now select a value of $p_C^*$. As before, we aim to equate the expected number of observed customer departures during level-$C$ busy periods in the IB and UWC models. Note, however, that unlike the exponential service case, we must now consider the service phase that is underway at the beginning of a level-$C$ busy period, $BP_C$. Similar to Equation (3.7), we define $q_{x,y}$ as the steady-state probability of the IB model being in state $(x, y)$ immediately prior to a customer arrival that initiates a level-$C$ busy period (i.e., an arrival that increases $X(t)$ from $C-1$ to $C$). It follows that at steady-state,

$$\begin{aligned} q_{C-1,y} &= \lim_{h\to 0} P((X(t),Y(t)) = (C-1,y)|X(t+h)=C) \\ &= \lim_{h\to 0} \frac{P(X(t+h)=C|(X(t),Y(t))=(C-1,y))P((X(t),Y(t))=(C-1,y))}{\sum_{m,n} P(X(t+h)=C|(X(t),Y(t))=(m,n))P((X(t),Y(t))=(m,n))} \\ &= \lim_{h\to 0} \frac{(\lambda h + o(h))\pi_{C-1,y}}{\sum_n (\lambda h + o(h))\pi_{C-1,n}} \\ &= \lim_{h\to 0} \frac{\lambda \pi_{C-1,y} + o(h)/h}{\sum_n \lambda \pi_{C-1,n} + o(h)/h} \\ &= \frac{\pi_{C-1,y}}{\underline{\pi}_{C-1}\underline{e}'}. \end{aligned} \tag{5.29}$$

Applying Equations (5.21) and (5.29), we define the modified initial probability row vector

$$\underline{\beta}^*_C = (q_{C-1,1}, q_{C-1,2}, \ldots, q_{C-1,b}) = \frac{\underline{\pi}_{C-1}}{\underline{\pi}_{C-1}\underline{e}'} = \frac{\underline{\beta}(\lambda I - \lambda \underline{e}'\underline{\beta} - B)^{-(C-1)}}{\underline{\beta}(\lambda I - \lambda \underline{e}'\underline{\beta} - B)^{-(C-1)}\underline{e}'}. \tag{5.30}$$

It now follows that $BP_C$ will be identical in distribution to a busy period of a modified IB $M/PH/1$ queue where the first customer of a busy period has a service time with distribution $Ser^*_C \sim \mathrm{PH}_b(\underline{\beta}^*_C, B)$, but all future service times within the same busy period will be iid with the original $\mathrm{PH}_b(\underline{\beta}, B)$ distribution. We can calculate $\mathrm{E}[BP_C]$ by setting $Q_{0,1} = \lambda \underline{\beta}^*_C$ in Equation (5.11) and solving for the modified steady-state distribution which we will define as $\underline{\pi}^{*C} = (\pi^{*C}_{0,0}, \underline{\pi}^{*C}_1, \underline{\pi}^{*C}_2, \ldots)$. By Equation (1.10), it readily follows that

$$\mathrm{E}[BP_C] = \frac{1-\pi^{*C}_{0,0}}{\lambda \pi^{*C}_{0,0}}. \tag{5.31}$$

Following similar steps to the original analysis for the IB model, we can show that

$$\underline{\pi}^{*C}_i = \pi^{*C}_{0,0}\underline{\beta}^*_C \lambda^i (\lambda I - \lambda \underline{e}'\underline{\beta} - B)^{-i}, \ i \in \mathbb{Z}^+.$$

Using Equation (5.20), we can now solve for $\pi^{*C}_{0,0}$ through the normalization condition,

$$\begin{aligned} 1 = \underline{\pi}^{*C}\underline{e}' &= \pi^{*C}_{0,0} + \underline{\pi}^{*C}_1 (I-R)^{-1}\underline{e}' \\ &= \pi^{*C}_{0,0}\left(1 + \underline{\beta}^*_C R (I-R)^{-1}\underline{e}'\right) \\ &= \pi^{*C}_{0,0}\left(1 + \frac{\lambda(-\underline{\beta}^*_C B^{-1}\underline{e}')}{1-\lambda \mathrm{E}[Ser]}\right) \\ &= \pi^{*C}_{0,0}\left(1 + \frac{\lambda \mathrm{E}[Ser^*_C]}{1-\lambda \mathrm{E}[Ser]}\right), \end{aligned}$$

resulting in

$$\pi^{*C}_{0,0} = \frac{1-\lambda \mathrm{E}[Ser]}{1+\lambda \mathrm{E}[Ser^*_C] - \lambda \mathrm{E}[Ser]},$$

and by substituting into Equation (5.31), it is straightforward to show that

$$\mathrm{E}[BP_C^*] = \frac{\mathrm{E}[Ser_C^*]}{1 - \lambda \mathrm{E}[Ser]}. \tag{5.32}$$

We now recall the left-hand side of Equation (5.28), which we can rewrite via Equation (5.30) as

$$\underline{\beta} R^{C-1} \lambda(-(B + p_C^* \underline{B}_0' \underline{\beta})^{-1})\underline{e}' = \lambda^C (\underline{\beta}(\lambda I - \lambda \underline{e}' \underline{\beta} - B)^{-(C-1)} \underline{e}')(-\underline{\beta}_C^* (B + p_C^* \underline{B}_0' \underline{\beta})^{-1} \underline{e}'). \tag{5.33}$$

Note that the term $-\underline{\beta}_C^*(B + p_C^* \underline{B}_0' \underline{\beta})^{-1}\underline{e}'$ is simply the expected value of a $\mathrm{PH}_b(\underline{\beta}_C^*, B + p_C^* \underline{B}_0' \underline{\beta})$ random variable. This corresponds to a phase-type distribution with initial probability row vector $\underline{\beta}_C^*$ and rate matrix $B$ that restarts with initial probability row vector $\underline{\beta}$ every time it would reach absorption with probability $p_C^*$. Note that we can express

$$BP_C = Ser_C^* + \sum_{j=1}^{N_C^*} Ser_j, \tag{5.34}$$

where $\{Ser_j\}_{j=1}^{\infty}$ are the iid service times having distribution $Ser \sim \mathrm{PH}_b(\underline{\beta}, B)$ within a busy period after the first service $Ser_C^* \sim \mathrm{PH}_b(\underline{\beta}_C^*, B)$, and $N_C^*$ is some discrete random variable depending on $\lambda$, $C$, and the random service times. If we approximate $N_C^*$ by an independent geometric distribution having probability mass function (PMF) $P(N = n) = (p_C^*)^n(1 - p_C^*)$, $n \in \mathbb{N}$, then this would be distributionally equivalent to $\mathrm{PH}_b(\underline{\beta}_C^*, B + p_C^* \underline{B}_0' \underline{\beta})$ (using the convention $\sum_{j=1}^{0} Ser_j = 0$).

Taking the expectation of Equation (5.34) under this approximation, we have

$$\mathrm{E}[BP_C] = \mathrm{E}[Ser_C^*] + \frac{p_C^*}{1 - p_C^*}\mathrm{E}[Ser]. \tag{5.35}$$

Equating Equations (5.32) and (5.35) and solving for $p_C^*$, we set

$$p_C^* = \frac{\lambda \mathrm{E}[Ser_C^*]}{1 + \lambda \mathrm{E}[Ser_C^*] - \lambda \mathrm{E}[Ser]}. \tag{5.36}$$

Therefore, if we use this choice of $p_C^*$, Equation (5.33) becomes

$$\begin{aligned}
\underline{\beta} R^{C-1} \lambda(-(B + p_C^* \underline{B}_0' \underline{\beta})^{-1})\underline{e}' &= \lambda^C (\underline{\beta}(\lambda I - \lambda \underline{e}' \underline{\beta} - B)^{-(C-1)} \underline{e}')\mathrm{E}[BP_C] \\
&= \frac{\lambda^C}{1 - \lambda \mathrm{E}[Ser]} (\underline{\beta}(\lambda I - \lambda \underline{e}' \underline{\beta} - B)^{-(C-1)} \underline{e}')\mathrm{E}[Ser_C^*] \\
&= \frac{\lambda^C}{1 - \lambda \mathrm{E}[Ser]} (\underline{\beta}(\lambda I - \lambda \underline{e}' \underline{\beta} - B)^{-(C-1)} \underline{e}')(-\underline{\beta}_C^* B^{-1} \underline{e}') \\
&= \frac{\lambda^C}{1 - \lambda \mathrm{E}[Ser]} \underline{\beta}(\lambda I - \lambda \underline{e}' \underline{\beta} - B)^{-(C-1)}(-B^{-1}\underline{e}').
\end{aligned}$$

Substituting Equation (5.20) into the right-hand side of Equation (5.28), it becomes

$$\underline{\beta} R^C (I - R)^{-1} \underline{e}' = \frac{\lambda^C}{1 - \lambda \mathrm{E}[Ser]} \underline{\beta}(\lambda I - \lambda \underline{e}' \underline{\beta} \underline{e}' - B)^{-(C-1)}(-B^{-1}\underline{e}'). \tag{5.37}$$

Thus, we have shown that the choice of $p_C^*$ in Equation (5.36) satisfies Equation (5.20), and it will hold that

$$\pi_{0,0}^{\mathrm{UWC}} = 1 - \lambda \mathrm{E}[\mathit{Ser}] = \pi_{0,0},$$
$$\underline{\pi}_i^{\mathrm{UWC}} = (1 - \lambda \mathrm{E}[\mathit{Ser}])\underline{\beta}\lambda^i(\lambda I - \lambda \underline{e}'\underline{\beta} - B)^{-i} = \underline{\pi}_i,\ i = 1, 2, \ldots, C-1,$$

and

$$\underline{\pi}_C^{\mathrm{UWC}} = (1 - \lambda \mathrm{E}[\mathit{Ser}])\underline{\beta}\lambda^C(\lambda I - \lambda \underline{e}'\underline{\beta} - B)^{-(C-1)}(-(B + p_C^*\underline{B}_0'\underline{\beta})^{-1}), \tag{5.38}$$

which must satisfy $\underline{\pi}_C^{\mathrm{UWC}}\underline{e}' = \sum_{i=C}^{\infty} \underline{\pi}_i\underline{e}'$.

We close this subsection by confirming that we recover the results of Section 5.2 if we let $\underline{\beta} = 1$ and $B = -\mu$ (i.e., if $\mathit{Ser} \sim \mathrm{Exp}(\mu)$). First of all, Equation (5.30) clearly simplifies to $\underline{\beta}_C^* = 1 = \underline{\beta}$. Therefore, $\mathit{Ser}_C^*$ and $\mathit{Ser}$ have identical distributions, implying that $\mathrm{E}[\mathit{Ser}_C^*] = \mathrm{E}[\mathit{Ser}]$ and Equation (5.36) simplifies to $p_C^* = \lambda \mathrm{E}[\mathit{Ser}] = \lambda/\mu = \rho$. Thus, Equation (5.38) reduces to

$$\begin{aligned} \pi_C^{\mathrm{UWC}} &= (1-\rho)\lambda^C(\lambda - \lambda - (-\mu))^{-(C-1)}(-(-\mu + \rho(\mu))^{-1}) \\ &= (1-\rho)\rho^{C-1}\frac{\lambda}{\mu - \lambda} = \rho^C, \end{aligned}$$

as required.

**Remark 5.1.** We can obtain the corresponding $M/PH/1$ FB model results by setting $p_C^* = 0$ in the above analysis. From Equations (5.28) and (5.37), since

$$\begin{aligned} \underline{\beta}R^{C-1}\lambda(-B^{-1}\underline{e}') &= \lambda^C\underline{\beta}(\lambda I - \lambda\underline{e}'\underline{\beta} - B)^{-(C-1)}(-B^{-1}\underline{e}') \\ &= (1 - \lambda \mathrm{E}[\mathit{Ser}])\underline{\beta}R^C(I - R)^{-1}\underline{e}' \\ &< \underline{\beta}R^C(I-R)^{-1}\underline{e}', \end{aligned}$$

it follows that $\pi_{0,0}^{\mathrm{FB}} > \pi_{0,0}$, and hence by Equation (5.26),

$$\underline{\pi}_i^{\mathrm{FB}} = \pi_{0,0}^{\mathrm{FB}}\underline{\beta}\lambda^i(\lambda I - \lambda\underline{e}'\underline{\beta} - B)^{-i} = \frac{\pi_{0,0}^{\mathrm{FB}}}{\pi_{0,0}} \cdot \underline{\pi}_i > \underline{\pi}_i,\ i = 1, 2, \ldots, C-1. \tag{5.39}$$

Interestingly, since Equation (5.39) confirms that $\underline{\pi}_{C-1}^{\mathrm{FB}}$ is proportional to $\underline{\pi}_{C-1}$, this implies that we can express Equation (5.30) in terms of the FB steady-state probabilities to obtain

$$\underline{\beta}_C^* = \frac{\underline{\pi}_{C-1}}{\underline{\pi}_{C-1}\underline{e}'} = \frac{\underline{\pi}_{C-1}^{\mathrm{FB}}}{\underline{\pi}_{C-1}^{\mathrm{FB}}\underline{e}'}.$$

## 5.6 $M/PH/1 + M$ Queue

Suppose now that individual customers not currently receiving service in the $M/PH/1$ queue from Section 5.5 are at risk of reneging according to iid $\mathrm{Exp}(\alpha)$ impatience timers (as in Section 1.2.6). This too may be modelled by a CTMC $\{(X(t), Y(t)), t \geq 0\}$ with the same interpretations as previously described. This CTMC is still a QBD whose infinitesimal generator takes the form of Equation (5.10), with non-zero matrix blocks of the form

$$\begin{array}{rclrclrcl} & & & Q_{0,0} &=& -\lambda, & Q_{0,1} &=& \lambda\underline{\beta}, \\ Q_{1,0} &=& \underline{B}_0', & Q_{1,1} &=& B - \lambda I, & Q_{1,2} &=& \lambda I, \\ Q_{i,i-1} &=& \underline{B}_0'\underline{\beta} + (i-1)\alpha I, & Q_{i,i} &=& B - (\lambda + (i-1)\alpha)I, & Q_{i,i+1} &=& \lambda I,\ i = 2, 3, \ldots. \end{array}$$

As $Q_{i,j}$ now depend on $i$ for $j = i-1, i$, this is a level-dependent QBD, which requires the analytical approach covered in Section 1.2.6 to solve for the steady-state probabilities. Recall that this numerical algorithm in fact calculates the steady-state distribution of the FB model approximation for a given truncation level $C$, and will converge to that of the IB model as $C \to \infty$.

For the UWC model where we truncate at level $C$, we must modify our approach due to the presence of reneging. In the $M/PH/1$ queue, the only way a customer could depart the system was through the completion of a service, after which the time until the next observed departure would have an iid distribution (i.e., a new service phase is always selected according to the probability vector $\underline{\beta}$). In the $M/PH/1+M$ queue, while we reinitialize the service phase after service completions, the current service phase is unchanged if we observe a departure due to impatience. Therefore, the random time intervals between observed departures are no longer iid and we cannot make a similar breakdown of a level-$C$ busy period as in Equation (5.34). That is, we are unable to directly obtain the expected number of observed departures from the expected duration of a level-$C$ busy period. Recall that in the $M/M/1+M$ queue, however, this was not a concern due to the existence of only a single service phase.

### 5.6.1 $M/PH/1+M$ Queue: UWC Version 1

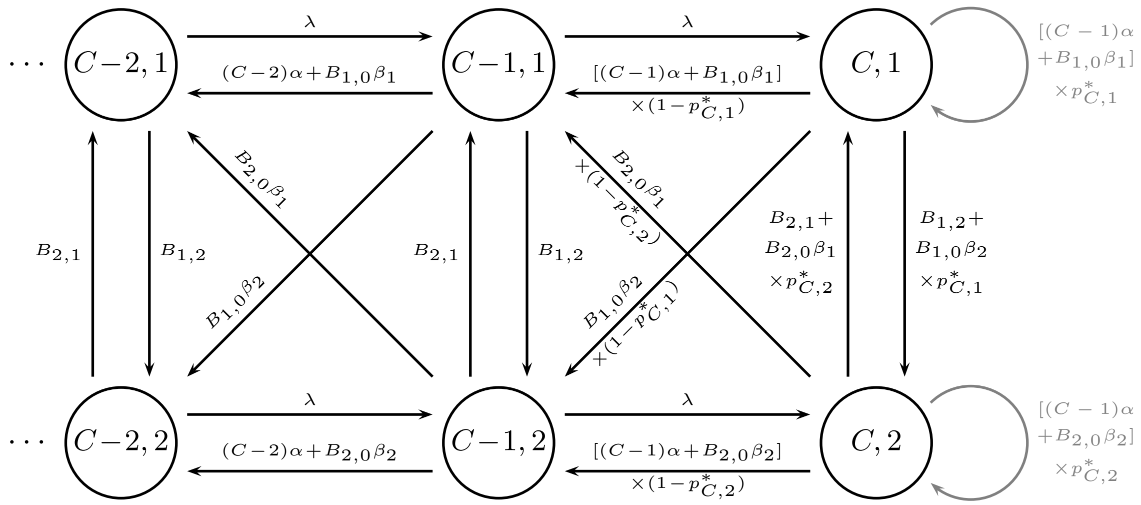

Figure 5.1: State transition diagram near truncation level $C$ for a UWC model of a $M/PH/1+M$ queue with two service phases.

We now propose two versions of the UWC approximation to tackle this harder problem. For UWC version 1, we obtain an analytic approximation that is comparable in computational complexity to our previous results, in that it does not require substantial additional computations relative to analyzing the FB model. To illustrate, we consider the UWC model of a $M/PH/1+M$ queue having two service phases. We visualize the state transition diagram of this model for states near level $C$ in Figure 5.1, where we denote the absorption rate out of the $j^{\text{th}}$ phase of *Ser* by $B_{j,0} = (\underline{B}'_0)_j$. While in state $(C,1)$, an observed departure will decrease the queue length with probability $1 - p^*_{C,1}$, the CTMC will remain in state $(C, 1)$ with probability

$$\frac{(C-1)\alpha + B_{1,0}\beta_1}{(C-1)\alpha + B_{1,0}} \cdot p^*_{C,1},$$

while the CTMC will transition to state $(C, 2)$ with probability

$$\frac{B_{1,0}\beta_2}{(C-1)\alpha + B_{1,0}} \cdot p^*_{C,1}.$$

That is, the UWC approximation will respond to departures in the same way in this model as in a $M/M/1 + M$ model with service rate $B_{1,0}$, with the exception of service completions that reinitialize the service phase in a *different* phase.

As $C \to \infty$, the process of observed customer departures while at level $C$ will become dominated by reneging, reducing the probability of an observed departure that would change the service phase but not decrement the queue length. Thus, in terms of the UWC behaviour, we approximate being in state $(C, j)$ as if in state $C$ of a $M/M/1 + M$ model with service rate $\mu$ equal to $B_{j,0}$. In the $M/M/1 + M$ queue, the probability $p^*_C$ in Equation (5.8) was optimal for estimating the probability of requiring at least one more observed departure to lower the level of the CTMC to $C - 1$. Therefore, upon observing a departure while in state $(C, j)$, we elect to let

$$p^*_{C,j} = 1 - \left( \sum_{k=1}^{\infty} \frac{\lambda^{k-1}(B_{j,0} + (C-1)\alpha)}{\prod_{n=0}^{k-1}(B_{j,0} + (C-1+n)\alpha)} \right)^{-1}, \; j = 1, 2, \ldots, b, \qquad (5.40)$$

be the probability of having one or more unobserved customers present in the system.

In addition to service completions that result in a change of service phase but not a reduction in the number of observed customers, there is an independent competing $\text{Exp}(B_{1,2})$ timer whose completion would result in a transition to state $(C, 2)$. When transitioning to $(C, 2)$ in either case, transitioning from $(C, 1)$ is treated in an identical fashion as transitioning from $(C-1, 2)$, and the model is now subject to UWC probability $p^*_{C,2}$. Thus, any information concerning how long the system has remained in level $C$ is effectively lost. We may interpret this as removing any unseen waiting customers present in the system at that point and starting a new level-$C$ busy period in an $M/M/1+M$ queue with service rate $B_{2,0}$. This intuition generalizes logically to any number of service phases, $b$. Therefore, while $p^*_{C,1}, p^*_{C,2}, \ldots, p^*_{C,b}$ will shift some steady-state probability mass to the truncation level $C$, they will underestimate the true expected number of required customer departures to transition to level $C - 1$ due to the removal of unseen waiting customers when the service phase (but not the number of observed customers) changes. While the gain in accuracy is not as great as in the simpler models, it will still outperform the standard FB model.

Additionally, unlike in Section 5.5, the equations for $\underline{\pi}_i^{\text{UWC}}$ will recursively depend on the form of $Q^{\text{UWC}}_{C,C}$ through

$$R_C = -Q_{C-1,C}(Q_{C,C})^{-1},$$

and

$$R_j = -Q_{j-1,j}(Q_{j,j} + R_{j+1}Q_{j+1,j})^{-1}, \; j \in \mathbb{Z}^+,$$

whereas previously they only depended on the value of $\underline{\pi}_C^{\text{UWC}}\underline{e}'$ through the normalization condition in Equation (5.27) used to obtain $\pi^{\text{UWC}}_{0,0}$. Therefore, as we will see in Tables 5.1

and 5.2, the less precise UWC approximation used in $Q^{\text{UWC}}_{C,C}$ and $Q^{\text{UWC}}_{C,C-1}$ may cause slight irregularities in levels near the buffer, where the dependency on $R_C$ is largest. However, these irregularities within a given level vanish as we increase $C$ and the distance between that level and the truncation level grows.

However, note that if $B_{0,1} = B_{0,2} = \cdots = B_{0,b}$, then $p^*_{C,1} = p^*_{C,2} = \cdots = p^*_{C,b}$ and no accuracy in the UWC approximation is lost. These choices of $p^*_{C,j}$ will also reduce to the $p^*_C$ of the $M/M/1+M$ UWC model if we assume exponentially distributed service times, as required. Moreover, as the proportion of departures due to reneging will increase and fewer instances of lost unobserved customers will occur as we increase $C$, the accuracy of the UWC approximation itself will also improve with larger $C$. Thus, like the other UWC models, the UWC steady-state distribution will converge to the IB steady-state distribution as $C \to \infty$.

### 5.6.2 $M/PH/1+M$ Queue: UWC Version 2

We consider a second version of UWC for this particular model. Rather than applying results from a simpler model, we apply phase-type theory to calculate the PMF of $N^*_C$ directly, from which we can obtain its expected value directly and use it to set a single UWC probability $p^*_C$. As in the analysis of Section 5.5, the initial service phase of the level-$C$ busy period matters, and like Equation (5.30) we would find that

$$\underline{\beta}^*_C = \frac{\underline{\pi}_{C-1}}{\underline{\pi}_{C-1}\underline{e}'}.$$

Unfortunately, we do not have a precise analytic solution of $\underline{\pi}_{C-1}$ from the IB model. We did, however, remark that for the $M/PH/1$ model, these IB steady-state probabilities may be replaced by those from the FB model with no loss of accuracy. While this is not the case for the $M/PH/1+M$ model, we still propose to use $\underline{\pi}^{\text{FB}}_{C-1}$ in place of $\underline{\pi}_{C-1}$, and let us denote this approximated initial probability row vector by $\hat{\underline{\beta}}^*$. As we will see in Tables 5.1 and 5.2, while we do not end up obtaining exact steady-state probabilities for levels below $C$, this approximation works very well. Note that this does imply that we must calculate the steady-state probabilities of the FB model prior to those of this version of the UWC model, effectively doubling our computational requirement (but still not requiring us to expand the considered state space).

Reusing $D$ to denote the level of truncation for a FB model approximation, we may model the number of customers beyond level $C$ (up until the next observed departure) by an absorbing CTMC having infinitesimal generator matrix

$$Q = \begin{bmatrix} Q_{TT} & Q_{TA} \\ \mathbf{0} & I \end{bmatrix},$$

where we let $\Delta = B - (\lambda + (C-1)\alpha)I_b$, $\Delta_A = \underline{B}'_0\underline{\beta} + (C-1)\alpha I_b$,

$$Q_{TT} = \begin{array}{c|cccccc|}
 & 0 & 1 & 2 & \cdots & D-1 & D \\
\hline
0 & \Delta & \lambda I_b & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\
1 & \alpha I_b & \Delta - \alpha I_b & \lambda I_b & \ddots & \mathbf{0} & \mathbf{0} \\
2 & \mathbf{0} & 2\alpha I_b & \Delta - 2\alpha I_b & \ddots & \mathbf{0} & \mathbf{0} \\
\vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\
D-1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \Delta - (D-1)\alpha I_b & \lambda I_b \\
D & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & D\alpha I_b & \Delta - (D\alpha - \lambda)I_b
\end{array},$$

and

$$Q_{TA} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ D-1 \\ D \end{array} \begin{array}{c} \begin{array}{cccccc} 0^* & 1^* & \cdots & (D-2)^* & (D-1)^* & -1^* \end{array} \\ \left[ \begin{array}{cccccc} \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \Delta_A \underline{e}' \\ \Delta_A & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \underline{0}'_b \\ \mathbf{0} & \Delta_A & \ddots & \mathbf{0} & \mathbf{0} & \underline{0}'_b \\ \vdots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \Delta_A & \mathbf{0} & \underline{0}'_b \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \Delta_A & \underline{0}'_b \end{array} \right] \end{array}.$$

This CTMC applies a FB approximation to the unobserved portion of the queue (considering an effective total queue length $C + D$). If it is absorbed into state $(i^*, j)$, $i \in \{0, 1, \ldots, D-1\}$, $j \in \{1, 2, \ldots, b\}$, then the queue length does not decrease after the next observed departure. After an unobserved customer immediately joins the observed portion of the queue, there are $i$ unobserved customers in the system, and the next service time begins in phase $j$. If it is absorbed into state $-1^*$, then there were no unobserved customers and the observed queue length will decrement.

Given the initial probability row vector $\hat{\underline{\beta}}^*$, if we let $D^*$ be a set of dummy absorption states (which cannot actually be observed) and define

$$Q^*_{TA} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ D-1 \\ D \end{array} \begin{array}{c} \begin{array}{cccccc} 0^* & 1^* & \cdots & (D-2)^* & (D-1)^* & D^* \end{array} \\ \left[ \begin{array}{cccccc} \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \underline{0}'_b \underline{0}_b \\ \Delta_A & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \Delta_A & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \Delta_A & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \Delta_A & \mathbf{0} \end{array} \right] \end{array},$$

while we let the right-most column of $Q_{TA}$ be denoted by

$$\underline{Q}'_{-1^*} = \begin{bmatrix} \Delta_A \underline{e}' \\ \underline{0}' \end{bmatrix},$$

then applying the absorbing CTMC theory from the Appendix, we know that by Equation (A.17), the probability that the queue length will not decrement after the first observed departure is

$$P(N^*_C \geq 1) = \begin{bmatrix} \hat{\underline{\beta}}^* & \underline{0} \end{bmatrix} (-Q^{-1}_{TT}) Q^*_{TA} \underline{e}',$$

and the probability that the length will decrement is

$$P(N^*_C = 0) = \begin{bmatrix} \hat{\underline{\beta}}^* & \underline{0} \end{bmatrix} (-Q^{-1}_{TT}) \underline{Q}'_{-1^*}.$$

In fact, we can use the knowledge of the absorption state to initialize the time until the next observed departure without losing track of the number of unobserved customers. That is,

$$P(N^*_C = n) = \begin{bmatrix} \hat{\underline{\beta}}^* & \underline{0} \end{bmatrix} \left[ (-Q^{-1}_{TT}) Q^*_{TA} \right]^n (-Q^{-1}_{TT}) \underline{Q}'_{-1^*}, \; n \in \mathbb{N}.$$

From here, we evaluate $\mathrm{E}[N^*_C]$ and select a UWC probability that equates

$$\mathrm{E}[N^*_C] = \frac{p^*_C}{1 - p^*_C},$$

or equivalently,

$$p^*_{C,j} = p^*_C = \frac{\mathrm{E}[N^*_C]}{1 + \mathrm{E}[N^*_C]},\ j = 1, 2, \ldots, b.$$

Note that this is an approximation for a given choice of $D \in \mathbb{Z}^+$. As such, a large enough $D$ should be selected such that this FB approximation approaches that of the true IB model. For the calculations within this subsection, we used $D = 40$.

### 5.6.3 $M/PH/1+M$ Queue: Comparing UWC Versions

Defining $\underline{p}^*_C = (p^*_{C,1}, p^*_{C,2}, \ldots, p^*_{C,b})$ and letting $D = C$ in the level-dependent QBD algorithm, we let the non-zero QBD blocks of the UWC model be given by $Q^{\mathrm{UWC}}_{i,j} = Q_{i,j}$, $i = 0, 1, \ldots, C-1$, $j = 0, 1, \ldots, C$,

$$\begin{aligned} Q^{\mathrm{UWC}}_{C,C-1} &= (I - \mathrm{diag}(\underline{p}^*_C))Q_{C,C-1} \\ &= (I - \mathrm{diag}(\underline{p}^*_C))(\underline{B}'_0\underline{\beta} + (C-1)\alpha I), \end{aligned} \tag{5.41}$$

and

$$\begin{aligned} Q^{\mathrm{UWC}}_{C,C} &= Q_{C,C} + \lambda I + \mathrm{diag}(\underline{p}^*_C)Q_{C,C-1} \\ &= B + \mathrm{diag}(\underline{p}^*_C)\underline{B}'_0\underline{\beta} - (I - \mathrm{diag}(\underline{p}^*_C))(C-1)\alpha I. \end{aligned} \tag{5.42}$$

In Tables 5.1 and 5.2, we illustrate the relative efficiency gains of both versions of this UWC model over the FB model. We apply the level-dependent QBD algorithm to approximate the IB model steady-state distribution $\underline{\pi}$ using a truncation level of 1000, as well as to calculate $\underline{\pi}^{\mathrm{UWC}}$ and $\underline{\pi}^{\mathrm{FB}}$ for $C = 3, 7$. We let $\lambda = 0.9$, $\alpha = 0.1$, and consider the following service time distributions:

- ($\mathrm{E}_2$) Erlang-2 with $\mathrm{E}[Ser] = 1$ and $\mathrm{Var}(Ser) = 0.5$:

$$Ser \sim \mathrm{PH}_2\left(\underline{\beta} = (1, 0), B = \begin{bmatrix} -2 & 2 \\ 0 & -2 \end{bmatrix}\right).$$

- ($\mathrm{E}^f_2$) Erlang-2 with feedback with $\mathrm{E}[Ser] = 1$ and $\mathrm{Var}(Ser) = 0.75$:

$$Ser \sim \mathrm{PH}_2\left(\underline{\beta} = (1, 0), B = \begin{bmatrix} -4 & 4 \\ 2 & -4 \end{bmatrix}\right).$$

- ($\mathrm{C}^f_2$) Coxian-2 with feedback with $\mathrm{E}[Ser] = 1$, $\mathrm{Var}(Ser) = 1.5$:

$$Ser \sim \mathrm{PH}_2\left(\underline{\beta} = (0.5, 0.5), B = \begin{bmatrix} -\left(\frac{8+4\sqrt{3}}{7+4\sqrt{3}}\right) & \frac{4+2\sqrt{3}}{7+4\sqrt{3}} \\ 4 + 2\sqrt{3} & -(8 + 4\sqrt{3}) \end{bmatrix}\right).$$

- ($\mathrm{H}_2$) Hyperexponential-2 with $\mathrm{E}[Ser] = 1$ and $\mathrm{Var}(Ser) = 2$:

$$Ser \sim \mathrm{PH}_2\left(\underline{\beta} = (0.5, 0.5), B = \begin{bmatrix} -(2 + \sqrt{2}) & 0 \\ 0 & -(2 - \sqrt{2}) \end{bmatrix}\right).$$

While it is clear that we do not recover $\underline{\pi}_i^{\text{UWC}} = \underline{\pi}_i$, $i = 0, 1, \ldots, C-1$, UWC version 2 results in approximations that are very close and note that (for a given $C$) UWC version 1 still gives a better overall fit of the true steady-state distribution than the FB model at these levels. Letting $\text{E}[X]$ denote the expected queue length at steady state for a given model, UWC version 2 provides the best estimates at all values of $C$ followed by version 1 and then the FB model (note that they will all converge to $\text{E}[X^{\text{IB}}]$ as $C \to \infty$). The UWC probability vectors $\underline{p}_C^*$ are also provided. Note that these probabilities are identical for version 1 under $\text{E}_2$ and $\text{E}_2^f$ service time distributions, since they have equal column vectors of absorption rates.

For UWC version 1, if we compare the four service time distributions and use the approximated IB model empty queue probabilities $\pi_0$ as a benchmark, this UWC approximation appears to work the best for $\text{H}_2$. Despite having the largest $\pi_0$, its $\pi_0^{\text{UWC}}$s have the smallest amount of error. This is intuitive, as $\text{H}_2$ does not permit service phase transitions without service completions (and hence, it acts most similar to the exponential distribution out of those considered). In contrast, the $\text{E}_2$ distribution will always observe one such transition, while the $\text{C}_2^f$ and $\text{E}_2^f$ distributions will have an expected value of two and four phase transitions between service completions, respectively.

The error observed in $\pi_0^{\text{UWC}}$ for the other three distributions when $C = 7$ is comparable. For smaller $C$, $\text{C}_2^f$ has the smallest error and the largest $\pi_0$ of the remaining three service time distributions, despite the fact that it observes twice as many phase transitions on average, relative to $\text{E}_2$. Therefore, the presence of absorbing rates equalling zero (e.g., $B_{1,0} = 0$) also appears to have a slight negative impact on the efficacy of UWC. Finally, since $\text{E}[X^{\text{UWC}}]$ at $C = 3$ for $\text{E}_2$ is larger than that for $\text{E}_2^f$, despite $\text{E}_2^f$ having the larger $\text{E}[X^{\text{IB}}]$, we can conclude that the $\text{E}_2$ has a slight edge in UWC performance over $\text{E}_2^f$ due to the latter's larger number of expected phase transitions.

We must also point out that it is possible to observe $\pi_{i,j}^{\text{UWC}} < \pi_{i,j}$. For example, at $i = 6$, $j = 2$, $C = 7$, and $\text{H}_2$ service, we have $\pi_{4,2}^{\text{UWC}} = 0.0408$ while $\pi_{4,2} = 0.0419$. Being only one level below the truncation level, this is an illustration of the possible irregularities mentioned previously. However, as if we further increase $C$, the distance between level 6 and the truncation increases and we observe $\pi_{4,2}^{\text{UWC}} = 0.0418$ for $C = 8$ and $\pi_{4,2}^{\text{UWC}} = 0.0419$ for $C = 9$. In either of these cases, UWC version 1 provides a closer estimate than FB, so despite not being perfect, it would clearly be preferable to use over FB for any of these service time distributions at a given $C$.

For UWC version 2, the performance for either distribution is notably better than that of version 1. Interestingly, its worst performance appears to be for the $\text{H}_2$ distribution, in contrast to version 1. It is also possible to observe underestimation of the true steady-state probabilities (e.g., $i = 0$, $C = 3$, $\text{H}_2$ service). If one can afford the extra computation time, it is clearly preferable to use version 2. However, note that its gains relative to version 1 are much smaller for the larger value of $C$, where a larger proportion of observed departures during a level-$C$ busy period are caused by reneging. Note also that in the case of exponential service, both versions of UWC will in fact result in the same optimal $p_C^*$, and so the analytic formula should be used.

Table 5.1: Steady-state probabilities and expected queue lengths for UWC versions 1 and 2, FB, and IB $M/PH/1+M$ queueing models, under $C=3,7$, $\lambda=0.9$, $\alpha=0.1$, and $\mathrm{E}_2$ or $\mathrm{E}_2^f$ service time distributions.

| $\mathrm{E}_2$ | | $C=3$ | | | $C=7$ | | |
|---|---|---|---|---|---|---|---|
| | UWC ver. 1 | UWC ver. 2 | FB | UWC ver. 1 | UWC ver. 2 | FB | IB |
| $\underline{\pi}_0$ | 0.2541 | 0.2263 | 0.2886 | 0.2269 | 0.2263 | 0.2293 | 0.2262 |
| $\underline{\pi}_1$ | $(0.1602, 0.1144)$ | $(0.1426, 0.1018)$ | $(0.1819, 0.1299)$ | $(0.1430, 0.1021)$ | $(0.1426, 0.1018)$ | $(0.1446, 0.1032)$ | $(0.1426, 0.1018)$ |
| $\underline{\pi}_2$ | $(0.1083, 0.1125)$ | $(0.0969, 0.1002)$ | $(0.1234, 0.1277)$ | $(0.0972, 0.1004)$ | $(0.0969, 0.1001)$ | $(0.0982, 0.1015)$ | $(0.0969, 0.1001)$ |
| $\underline{\pi}_3$ | $(0.1045, 0.1461)$ | $(0.1473, 0.1849)$ | $(0.0505, 0.0981)$ | $(0.0638, 0.0751)$ | $(0.0636, 0.0748)$ | $(0.0644, 0.0759)$ | $(0.0636, 0.0748)$ |
| $\underline{\pi}_4$ | - | - | - | $(0.0391, 0.0492)$ | $(0.0390, 0.0491)$ | $(0.0395, 0.0497)$ | $(0.0390, 0.0491)$ |
| $\underline{\pi}_5$ | - | - | - | $(0.0223, 0.0294)$ | $(0.0222, 0.0293)$ | $(0.0225, 0.0297)$ | $(0.0222, 0.0293)$ |
| $\underline{\pi}_6$ | - | - | - | $(0.0116, 0.0163)$ | $(0.0118, 0.0162)$ | $(0.0119, 0.0164)$ | $(0.0118, 0.0162)$ |
| $\underline{\pi}_7$ | - | - | - | $(0.0095, 0.0140)$ | $(0.0107, 0.0155)$ | $(0.0041, 0.0089)$ | $(0.0059, 0.0083)$ |
| $\mathrm{E}[X]$ | 1.4677 | 1.6352 | 1.2597 | 2.0010 | 2.0154 | 1.9473 | 2.0362 |
| $\underline{p}_C^*$ | $(0.9950, 0.3820)$ | $(0.5934, 0.5934)$ | $(0, 0)$ | $(0.8970, 0.3278)$ | $(0.4604, 0.4604)$ | $(0, 0)$ | - |

| $\mathrm{E}_2^f$ | | $C=3$ | | | $C=7$ | | |
|---|---|---|---|---|---|---|---|
| | UWC ver. 1 | UWC ver. 2 | FB | UWC ver. 1 | UWC ver. 2 | FB | IB |
| $\underline{\pi}_0$ | 0.2655 | 0.2342 | 0.3009 | 0.2351 | 0.2342 | 0.2382 | 0.2342 |
| $\underline{\pi}_1$ | $(0.1437, 0.1195)$ | $(0.1267, 0.1054)$ | $(0.1628, 0.1354)$ | $(0.1272, 0.1058)$ | $(0.1267, 0.1054)$ | $(0.1289, 0.1072)$ | $(0.1267, 0.1054)$ |
| $\underline{\pi}_2$ | $(0.1036, 0.1079)$ | $(0.0916, 0.0951)$ | $(0.1177, 0.1222)$ | $(0.0920, 0.0955)$ | $(0.0916, 0.0951)$ | $(0.0932, 0.0968)$ | $(0.0916, 0.0951)$ |
| $\underline{\pi}_3$ | $(0.1199, 0.1399)$ | $(0.1645, 0.1824)$ | $(0.0690, 0.0919)$ | $(0.0652, 0.0708)$ | $(0.0649, 0.0705)$ | $(0.0661, 0.0717)$ | $(0.0649, 0.0705)$ |
| $\underline{\pi}_4$ | - | - | - | $(0.0430, 0.0476)$ | $(0.0428, 0.0474)$ | $(0.0436, 0.0482)$ | $(0.0428, 0.0474)$ |
| $\underline{\pi}_5$ | - | - | - | $(0.0263, 0.0296)$ | $(0.0262, 0.0295)$ | $(0.0267, 0.0300)$ | $(0.0262, 0.0295)$ |
| $\underline{\pi}_6$ | - | - | - | $(0.0149, 0.0172)$ | $(0.0150, 0.0171)$ | $(0.0152, 0.0174)$ | $(0.0150, 0.0171)$ |
| $\underline{\pi}_7$ | - | - | - | $(0.0138, 0.0160)$ | $(0.0155, 0.0180)$ | $(0.0072, 0.0096)$ | $(0.0080, 0.0092)$ |
| $\mathrm{E}[X]$ | 1.4656 | 1.6465 | 1.2607 | 2.0587 | 2.0784 | 1.9931 | 2.1077 |
| $\underline{p}_C^*$ | $(0.9950, 0.3820)$ | $(0.6129, 0.6129)$ | $(0, 0)$ | $(0.8970, 0.3278)$ | $(0.4857, 0.4857)$ | $(0, 0)$ | - |

Table 5.2: Steady-state probabilities and expected queue lengths for UWC versions 1 and 2, FB, and IB $M/PH/1+M$ queueing models, under $C=3,7$, $\lambda=0.9$, $\alpha=0.1$, and $\mathrm{C}_2^f$ or $\mathrm{H}_2$ service time distributions.

| $\mathrm{C}_2^f$ | | $C=3$ | | | $C=7$ | | |
|---|---|---|---|---|---|---|---|
| | UWC ver. 1 | UWC ver. 2 | FB | UWC ver. 1 | UWC ver. 2 | FB | IB |
| $\underline{\pi}_0$ | 0.2772 | 0.2559 | 0.3355 | 0.2568 | 0.2560 | 0.2630 | 0.2560 |
| $\underline{\pi}_1$ | $(0.1899, 0.0198)$ | $(0.1754, 0.0183)$ | $(0.2300, 0.0239)$ | $(0.1760, 0.0183)$ | $(0.1755, 0.0183)$ | $(0.1803, 0.0188)$ | $(0.1755, 0.0183)$ |
| $\underline{\pi}_2$ | $(0.1627, 0.0113)$ | $(0.1539, 0.0101)$ | $(0.2029, 0.0132)$ | $(0.1543, 0.0101)$ | $(0.1538, 0.0101)$ | $(0.1580, 0.0104)$ | $(0.1538, 0.0101)$ |
| $\underline{\pi}_3$ | $(0.3218, 0.0174)$ | $(0.3649, 0.0215)$ | $(0.1871, 0.0074)$ | $(0.1227, 0.0075)$ | $(0.1223, 0.0075)$ | $(0.1257, 0.0077)$ | $(0.1223, 0.0075)$ |
| $\underline{\pi}_4$ | - | - | - | $(0.0903, 0.0054)$ | $(0.0901, 0.0054)$ | $(0.0925, 0.0055)$ | $(0.0901, 0.0054)$ |
| $\underline{\pi}_5$ | - | - | - | $(0.0618, 0.0036)$ | $(0.0617, 0.0036)$ | $(0.0635, 0.0037)$ | $(0.0617, 0.0036)$ |
| $\underline{\pi}_6$ | - | - | - | $(0.0389, 0.0023)$ | $(0.0396, 0.0022)$ | $(0.0409, 0.0023)$ | $(0.0396, 0.0022)$ |
| $\underline{\pi}_7$ | - | - | - | $(0.0494, 0.0025)$ | $(0.0512, 0.0028)$ | $(0.0267, 0.0011)$ | $(0.0238, 0.0013)$ |
| $\mathrm{E}[X]$ | 1.5751 | 1.6808 | 1.2695 | 2.2342 | 2.2479 | 2.1173 | 2.3042 |
| $\underline{p}_C^*$ | $(0.8355, 0.1157)$ | $(0.6592, 0.6592)$ | $(0, 0)$ | $(0.6555, 0.1101)$ | $(0.5331, 0.5331)$ | $(0, 0)$ | - |

| $\mathrm{H}_2$ | | $C=3$ | | | $C=7$ | | |
|---|---|---|---|---|---|---|---|
| | UWC ver. 1 | UWC ver. 2 | FB | UWC ver. 1 | UWC ver. 2 | FB | IB |
| $\underline{\pi}_0$ | 0.2791 | 0.2649 | 0.3472 | 0.2663 | 0.2657 | 0.2746 | 0.2658 |
| $\underline{\pi}_1$ | $(0.0483, 0.1473)$ | $(0.0458, 0.1401)$ | $(0.0600, 0.1839)$ | $(0.0460, 0.1408)$ | $(0.0459, 0.1405)$ | $(0.0475, 0.1452)$ | $(0.0460, 0.1406)$ |
| $\underline{\pi}_2$ | $(0.0247, 0.1300)$ | $(0.0224, 0.1289)$ | $(0.0285, 0.1740)$ | $(0.0227, 0.1289)$ | $(0.0227, 0.1286)$ | $(0.0234, 0.1328)$ | $(0.0227, 0.1286)$ |
| $\underline{\pi}_3$ | $(0.0325, 0.3380)$ | $(0.0394, 0.3584)$ | $(0.0071, 0.1993)$ | $(0.0142, 0.1081)$ | $(0.0142, 0.1079)$ | $(0.0147, 0.1115)$ | $(0.0142, 0.1079)$ |
| $\underline{\pi}_4$ | - | - | - | $(0.0095, 0.0843)$ | $(0.0095, 0.0841)$ | $(0.0098, 0.0870)$ | $(0.0095, 0.0841)$ |
| $\underline{\pi}_5$ | - | - | - | $(0.0063, 0.0612)$ | $(0.0062, 0.0613)$ | $(0.0064, 0.0637)$ | $(0.0063, 0.0613)$ |
| $\underline{\pi}_6$ | - | - | - | $(0.0042, 0.0408)$ | $(0.0039, 0.0420)$ | $(0.0037, 0.0449)$ | $(0.0039, 0.0419)$ |
| $\underline{\pi}_7$ | - | - | - | $(0.0044, 0.0621)$ | $(0.0051, 0.0625)$ | $(0.0008, 0.0341)$ | $(0.0023, 0.0269)$ |
| $\mathrm{E}[X]$ | 1.6166 | 1.6821 | 1.2680 | 2.3054 | 2.3153 | 2.1570 | 2.3922 |
| $\underline{p}_C^*$ | $(0.2404, 0.8119)$ | $(0.6787, 0.6787)$ | $(0, 0)$ | $(0.2173, 0.6360)$ | $(0.5630, 0.5630)$ | $(0, 0)$ | - |

## 5.7 $N$-Queue $M/PH/1+M$ Exhaustive Polling System

### 5.7.1 Model Assumptions

We consider a system of $N$ queues, $Q_1, Q_2, \ldots, Q_N$, which are visited in a cyclic order by a lone server. The server follows an exhaustive service discipline such that once the server visits a queue, they do not leave until it has emptied. If the server arrives to a queue and finds it to be empty, they immediately move on to the next queue. Let a class-$i$ switch-in time denote the amount of time that it takes the server to switch from $Q_{i-1}$ to $Q_i$ (where $Q_0$ represents $Q_N$). We assume that switch-in times are independent, and class-$i$ switch-in times follow a $\mathrm{PH}_{s_i}(\underline{\gamma}_i, S_i)$ distribution with column vector of absorption rates $\underline{S}_{0,i} = -S_i\underline{e}'$, $i = 1, 2, \ldots, N$. Furthermore, we assume that switch-in times are strictly positive in duration (i.e., $\underline{\gamma}_i\underline{e}' = 1$).

Each $Q_i$ has its own class of customers who arrive according to independent Poisson processes with parameters $\lambda_i$, $i = 1, 2, \ldots, N$. Class-$i$ customers are served according to a FCFS order within their queue, having independent service time requirements $Ser_i \sim \mathrm{PH}_{b_i}(\underline{\beta}_i, B_i)$ with column vector of absorption rates $\underline{B}_{0,i} = -B_i\underline{e}'$. Additionally, class-$i$ customers are assumed to have independent $\mathrm{Exp}(\alpha_i)$ impatience timers, and are at risk of reneging up until they reach the server. We may set $\alpha_i$ to be zero, in which case class-$i$ customers are patient and are not at risk of reneging.

We let the truncation of $Q_i$ be at queue length $C_i < \infty$. Define $p^*_{i,j}$ as the UWC probability applied to class $i$ (at queue length $C_i$) when the server is at $Q_i$ and is currently in phase $j$ of a customer's service time distribution, $j = 1, 2, \ldots, b_i$. Here, for ease of notation, we suppress the dependency of $p^*_{i,j}$ on $C_i$. When the server is not currently visiting $Q_i$ and there are $C_i$ observed class-$i$ customers in $Q_i$, we use UWC probability $p^*_{i,0}$.

For now, suppose that $\alpha_i > 0$, $i = 1, 2, \ldots, N$. If the server is not at $Q_i$, then $Q_i$ acts as an $M/M/\infty$ queue with $Ser \sim \mathrm{Exp}(\alpha_i)$. It is therefore a logical choice to apply Equation (5.9), and set

$$p^*_{i,0} = 1 - \frac{(\lambda_i/\alpha_i)^{C_i}}{C_i!}\left(e^{\lambda_i/\alpha_i} - \sum_{k=0}^{C_i-1}\frac{(\lambda_i/\alpha_i)^k}{k!}\right)^{-1}.$$

For $p^*_{i,j}$, $j = 1, 2, \ldots, b_i$, we elect to use analogues of UWC versions 1 and 2 from Section 5.6. When applying version 1, we use Equation (5.40) and set

$$p^*_{i,j} = 1 - \left(\sum_{k=1}^{\infty}\frac{\lambda^{k-1}(B_{j,0,i} + (C_i-1)\alpha_i)}{\prod_{n=0}^{k-1}(B_{j,0,i} + (C_i - 1 + n)\alpha_i)}\right)^{-1}, \ j = 1, 2, \ldots, b_i, \tag{5.43}$$

where $B_{j,0,i} = (\underline{B}_{0,i})_j$.

In order to apply version 2, we require an initial probability vector for the first service in the level-$C$ busy period for every class $i = 1, 2, \ldots, N$. Following a similar logic to what was used to derive Equation (5.29), the steady-state probability of the IB model initializing a level-$C_i$ busy period in service phase $y$, $y = 1, 2, \ldots, b_i$, is

$$\frac{\sum_{n_1,\ldots,n_{i-1},n_{i+1},\ldots,n_N}(\lambda_i\pi^{\mathrm{IB}}_{n_1,\ldots,n_{i-1},C_i-1,n_{i+1},\ldots,n_N,2i,y} + \underline{\pi}^{\mathrm{IB}}_{n_1,\ldots,n_{i-1},C_i,n_{i+1},\ldots,n_N,2i-1}\underline{S}'_{0,i}\beta_{i,y})}{\sum_{n_1,\ldots,n_{i-1},n_{i+1},\ldots,n_N}(\lambda_i\underline{\pi}^{\mathrm{IB}}_{n_1,\ldots,n_{i-1},C_i-1,n_{i+1},\ldots,n_N,2i}\underline{e}' + \underline{\pi}^{\mathrm{IB}}_{n_1,\ldots,n_{i-1},C_i,n_{i+1},\ldots,n_N,2i-1}\underline{S}'_{0,i})},$$

where $\beta_{i,y} = (\underline{\beta}_i)_y$. We now define the corresponding modified phase-type initial probability row vector

$$\underline{\beta}^*_i = \frac{\sum_{n_1,\ldots,n_{i-1},n_{i+1},\ldots,n_N}(\lambda_i\underline{\pi}^{\mathrm{IB}}_{n_1,\ldots,n_{i-1},C_i-1,n_{i+1},\ldots,n_N,2i} + \underline{\pi}^{\mathrm{IB}}_{n_1,\ldots,n_{i-1},C_i,n_{i+1},\ldots,n_N,2i-1}\underline{S}'_{0,i}\underline{\beta}_i)}{\sum_{n_1,\ldots,n_{i-1},n_{i+1},\ldots,n_N}(\lambda_i\underline{\pi}^{\mathrm{IB}}_{n_1,\ldots,n_{i-1},C_i-1,n_{i+1},\ldots,n_N,2i}\underline{e}' + \underline{\pi}^{\mathrm{IB}}_{n_1,\ldots,n_{i-1},C_i,n_{i+1},\ldots,n_N,2i-1}\underline{S}'_{0,i})}. \tag{5.44}$$

As we do not in general know the true IB steady-state probabilities, we again approximate this by using FB model steady-state probabilities and refer to the approximated vector as $\underline{\hat{\beta}}_i^*$. Given these probability vectors, we repeat the numerical procedure for UWC version 2 for every class to obtain UWC probabilities $p_i^*$, and then let each $p_{i,j}^* = p_i^*$, $j = 1, 2, \ldots, b_i$, $i = 1, 2, \ldots, N$. We will consider the possibility of one or more queues having patient customers (i.e., $\alpha_i = 0$) in Section 5.7.4.

### 5.7.2 State Space and Steady-State Probabilities

This $N$-queue system may be modelled by the CTMC

$$\{(X_1(t), X_2(t), \ldots, X_N(t), L(t), Y(t)), t \geq 0\},$$

where $X_i(t) \in \{0, 1, \ldots, C_i\}$ is the number of class-$i$ customers in the system, $i = 1, 2, \ldots, N$, $L(t) \in \{1, 2, \ldots, 2N-1, 2N\}$ denotes the location of the server, where $L(t) = 2i-1$ if the server is conducting a class-$i$ switch-in or $L(t) = 2i$ if they are serving class $i$, such that

$$L(t) \in \Omega_L(X_1(t), X_2(t), \ldots, X_N(t)) = \bigcup_{i=1}^{N} \Omega_L(X_i(t)),$$

where we define for $i = 1, 2, \ldots, N$,

$$\Omega_L(X_i(t)) = \begin{cases} \{2i-1\} & , \text{ if } X_i(t) = 0, \\ \{2i-1, 2i\} & , \text{ if } X_i(t) > 0, \end{cases}$$

and $Y(t)$ tracks the current service or switch-in phase, taking possible values depending on $L(t)$ as follows:

$$Y(t) \in \Omega_Y(L(t)) = \begin{cases} \{1, 2, \ldots, s_1\} & , \text{ if } L(t) = 1, \\ \{1, 2, \ldots, b_1\} & , \text{ if } L(t) = 2, \\ & \vdots \\ \{1, 2, \ldots, s_i\} & , \text{ if } L(t) = 2i-1, \\ \{1, 2, \ldots, b_i\} & , \text{ if } L(t) = 2i, \\ & \vdots \\ \{1, 2, \ldots, s_N\} & , \text{ if } L(t) = 2N-1, \\ \{1, 2, \ldots, b_N\} & , \text{ if } L(t) = 2N. \end{cases}$$

Letting $s = \sum_{i=1}^{N} s_i$, this CTMC has

$$s \prod_{i=1}^{N} (C_i + 1) + \sum_{j=1}^{N} b_j \prod_{i=1}^{N} (C_i + 1 - \delta_{i,j}) \tag{5.45}$$

total states.

Let $\pi_{n_1, n_2, \ldots, n_N, l, y}$ be the steady-state probability of observing the CTMC in state $(n_1, n_2, \ldots, n_N, l, y)$. As we are truncating $Q_i$ at $C_i$, $i = 1, 2, \ldots, N$, these are not IB model probabilities, but rather

UWC model probabilities by default, or FB model probabilities if we let every $p^*_{i,j} = 0$. For $i = 1, 2, \ldots, N$, we organize them into ordered row vectors as follows:

$$\underline{\pi}_{n_1,n_2,\ldots,n_N,l} = \begin{cases} (\pi_{n_1,n_2,\ldots,n_N,l,1}, \pi_{n_1,n_2,\ldots,n_N,l,2}, \ldots, \pi_{n_1,n_2,\ldots,n_N,l,s_i}) & \text{, if } l = 2i-1, \\ (\pi_{n_1,n_2,\ldots,n_N,l,1}, \pi_{n_1,n_2,\ldots,n_N,l,2}, \ldots, \pi_{n_1,n_2,\ldots,n_N,l,b_i}) & \text{, if } l = 2i,\ n_i \geq 1. \end{cases}$$

Next, these vectors are further sorted into

$$\underline{\pi}_{n_1,n_2,\ldots,n_N} = (\underline{\pi}^{[1]}_{n_1,n_2,\ldots,n_N}, \underline{\pi}^{[2]}_{n_1,n_2,\ldots,n_N}, \ldots, \underline{\pi}^{[N]}_{n_1,n_2,\ldots,n_N}),$$

where

$$\underline{\pi}^{[i]}_{n_1,n_2,\ldots,n_N} = \begin{cases} \underline{\pi}_{n_1,n_2,\ldots,n_N,2i-1} & \text{, if } n_i = 0, \\ (\underline{\pi}_{n_1,n_2,\ldots,n_N,2i-1}, \underline{\pi}_{n_1,n_2,\ldots,n_N,2i}) & \text{, if } n_i > 0. \end{cases}$$

We finally group these vectors into probability row vectors

$$\begin{aligned} \underline{\pi}_{n_1} &= (\underline{\pi}_{n_1,0}, \underline{\pi}_{n_1,1}, \ldots, \underline{\pi}_{n_1,C_2}), \\ \underline{\pi}_{n_1,n_2} &= (\underline{\pi}_{n_1,n_2,0}, \underline{\pi}_{n_1,n_2,1}, \ldots, \underline{\pi}_{n_1,n_2,C_3}), \\ &\vdots \\ \underline{\pi}_{n_1,n_2,\ldots,n_i} &= (\underline{\pi}_{n_1,n_2,\ldots,n_i,0}, \underline{\pi}_{n_1,n_2,\ldots,n_i,1}, \ldots, \underline{\pi}_{n_1,n_2,\ldots,n_i,C_{i+1}}),\ i = 1, 2, \ldots, N-1, \end{aligned}$$

such that $\underline{\pi} = (\underline{\pi}_0, \underline{\pi}_1, \ldots, \underline{\pi}_{C_1})$ is the combined probability row vector having $C_1 + 1$ levels. We can solve for these probabilities using the QBD specified in Section 5.7.3, applying the algorithm outlined in Section 1.2.6.

### 5.7.3 Infinitesimal Generator Matrix

Letting the value of $X_1(t)$ denote the level of the process, we now construct the generator blocks, $Q_{i,j}$, which contain all transition probabilities that result in the level changing from $i$ to $j$. To begin, we define

$$\lambda_{n_1,\ldots,n_N} = \sum_{i=1}^{N} \lambda_i (1 - \delta_{n_i,C_i}),$$

$$a^{[m,n]}_{n_1,\ldots,n_N} = \sum_{i=m}^{n} (s_i + (1 - \delta_{n_i,0}) b_i),\ 1 \leq m \leq n \leq N,$$

$$\underline{p}^*_i = (p^*_{i,1}, p^*_{i,2}, \ldots, p^*_{i,b_i}),$$

$$p^*_{i,n_i,l,y} = \begin{cases} p^*_{i,0} & \text{, if } n_i = C_i,\ l \neq 2i, \\ p^*_{i,y} & \text{, if } n_i = C_i,\ l = 2i, \\ 0 & \text{, otherwise,} \end{cases}$$

$$\alpha_{n_1,\ldots,n_N,l,y} = \sum_{i=1}^{N} \alpha_i (n_i - \delta_{l,2i})(1 - p^*_{i,n_i,l,y}),$$

$$\underline{\alpha}_{n_1,\ldots,n_N,l} = \begin{cases} (\alpha_{n_1,\ldots,n_N,l,1}, \alpha_{n_1,\ldots,n_N,l,2}, \ldots, \alpha_{n_1,\ldots,n_N,l,s_i}) & \text{, if } l = 2i-1, \\ (\alpha_{n_1,\ldots,n_N,l,1}, \alpha_{n_1,\ldots,n_N,l,2}, \ldots, \alpha_{n_1,\ldots,n_N,l,b_i}) & \text{, if } l = 2i, \end{cases}$$

and

$$B^*_{i,n_i} = B_i + \delta_{n_i,C_i}\mathrm{diag}(\underline{p}^*_i)\underline{B}'_{0,i}\underline{\beta}_i,\ i = 1, 2, \ldots, N.$$

We first construct blocks to track movements in $X_N(t)$, after which we will recursively build outwards to track all queue lengths. We achieve this by modelling changes of $X_j(t)$ for given values of $X_1(t), X_2(t), \ldots, X_{j-1}(t)$ using a QBD structure. These will be nested within each other, with the innermost QBDs describing $X_N(t)$. For $n_i = 0, 1, \ldots, C_i$, $i = 1, 2, \ldots, N-1$, we define

$$Q^{[N]}_{n_1,\ldots,n_{N-1}} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C_N\!-\!1 \\ C_N \end{array} \begin{array}{c} \begin{array}{cccccc} 0 & 1 & 2 & \cdots & C_N\!-\!1 & C_N \end{array} \\ \left[\begin{array}{cccccc} \Delta_{n_1,\ldots,n_{N-1},0} & (UD)^{[N]}_{n_1,\ldots,n_{N-1},0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ (LD)^{[N]}_{n_1,\ldots,n_{N-1},1} & \Delta_{n_1,\ldots,n_{N-1},1} & (UD)^{[N]}_{n_1,\ldots,n_{N-1},1} & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (LD)^{[N]}_{n_1,\ldots,n_{N-1},2} & \Delta_{n_1,\ldots,n_{N-1},2} & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \Delta_{n_1,\ldots,n_{N-1},c_N-1} & (UD)^{[N]}_{n_1,\ldots,n_{N-1},c_N-1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & (LD)^{[N]}_{n_1,\ldots,n_{N-1},c_N} & \Delta_{n_1,\ldots,n_{N-1},c_N} \end{array}\right] \end{array}.$$

Letting

$$\zeta_{n_1,\ldots,n_N,l} = \begin{cases} S_j - \lambda_{n_1,\ldots,n_N} I_{s_j} - \mathrm{diag}(\underline{\alpha}_{n_1,\ldots,n_N,2j-1}) & ,\ \text{if } l = 2j-1, \\ B^*_{j,n_j} - \lambda_{n_1,\ldots,n_N} I_{b_j} - \mathrm{diag}(\underline{\alpha}_{n_1,\ldots,n_N,2j}) & ,\ \text{if } l = 2j, \end{cases}$$

the main diagonal blocks of $Q^{[N]}_{n_1,\ldots,n_{N-1}}$ are

$$\Delta_{n_1,\ldots,n_{N-1},n_N} = \begin{bmatrix} \Delta^{[1]}_{n_1,\ldots,n_N} \\ \Delta^{[2]}_{n_1,\ldots,n_N} \\ \vdots \\ \Delta^{[N]}_{n_1,\ldots,n_N} \end{bmatrix},$$

where

$$\Delta^{[1]}_{n_1,\ldots,n_N} = \begin{cases} \begin{bmatrix} \zeta_{n_1,\ldots,n_N,1} & \underline{S}'_{0,1}\underline{\gamma}_2 & \underline{0}'_{s_1}\underline{0}_{a^{[2,N]}_{n_1,\ldots,n_N}-s_2} \end{bmatrix} & ,\ \text{if } n_1 = 0, \\ \\ \begin{bmatrix} \zeta_{n_1,\ldots,n_N,1} & \underline{S}'_{0,1}\underline{\beta}_1 & \underline{0}'_{s_1}\underline{0}_{a^{[2,N]}_{n_1,\ldots,n_N}} \\ \underline{0}'_{b_1}\underline{0}_{s_1} & \zeta_{n_1,\ldots,n_N,2} & \underline{0}'_{b_1}\underline{0}_{a^{[2,N]}_{n_1,\ldots,n_N}} \end{bmatrix} & ,\ \text{if } n_1 = 1, 2, \ldots, C_1, \end{cases}$$

while for $j = 2, 3 \ldots, N-1$,

$$\Delta^{[j]}_{n_1,\ldots,n_N} = \begin{cases} \begin{bmatrix} \underline{0}'_{s_j}\underline{0}_{a^{[1,j-1]}_{n_1,\ldots,n_N}} & \zeta_{n_1,\ldots,n_N,2j-1} & \underline{S}'_{0,j}\underline{\gamma}_{j+1} & \underline{0}'_{s_j}\underline{0}_{a^{[j+1,N]}_{n_1,\ldots,n_N}-s_{j+1}} \end{bmatrix} & ,\ \text{if } n_j = 0, \\ \\ \begin{bmatrix} \underline{0}'_{s_j}\underline{0}_{a^{[1,j-1]}_{n_1,\ldots,n_N}} & \zeta_{n_1,\ldots,n_N,2j-1} & \underline{S}'_{0,j}\underline{\beta}_j & \underline{0}'_{s_j}\underline{0}_{a^{[j+1,N]}_{n_1,\ldots,n_N}} \\ \underline{0}'_{b_j}\underline{0}_{a^{[1,j-1]}_{n_1,\ldots,n_N}} & \underline{0}'_{b_j}\underline{0}_{s_j} & \zeta_{n_1,\ldots,n_N,2j} & \underline{0}'_{b_j}\underline{0}_{a^{[j+1,N]}_{n_1,\ldots,n_N}} \end{bmatrix} & ,\ \text{if } n_j = 1, 2, \ldots, C_j, \end{cases}$$

and

$$\Delta^{[N]}_{n_1,\ldots,n_N} = \begin{cases} \begin{bmatrix} \underline{S}'_{0,N}\underline{\gamma}_1 & \underline{0}'_{s_N}\underline{0}_{a^{[1,N-1]}_{n_1,\ldots,n_N}-s_1} & \zeta_{n_1,\ldots,n_N,2N-1} \end{bmatrix} & ,\ \text{if } n_N = 0, \\ \\ \begin{bmatrix} \underline{0}'_{s_N}\underline{0}_{a^{[1,N-1]}_{n_1,\ldots,n_N}} & \zeta_{n_1,\ldots,n_N,2N-1} & \underline{S}'_{0,N}\beta_N \\ \underline{0}'_{b_N}\underline{0}_{a^{[1,N-1]}_{n_1,\ldots,n_N}} & \underline{0}'_{b_N}\underline{0}_{s_N} & \zeta_{n_1,\ldots,n_N,2N} \end{bmatrix} & ,\ \text{if } n_N = 1, 2, \ldots, C_N. \end{cases}$$

The upper diagonal blocks of $Q^{[N]}_{n_1,\ldots,n_{N-1}}$ are

$$(UD)^{[N]}_{n_1,\ldots,n_{N-1},n_N} = \left[\begin{array}{cc} \lambda_N I_{a^{[1,N-1]}_{n_1,\ldots,n_N}+s_N} & \underline{0}'_{a^{[1,N-1]}_{n_1,\ldots,n_N}+s_N}\underline{0}_{b_N} \end{array}\right]$$

for $n_N = 0$ and

$$(UD)^{[N]}_{n_1,\ldots,n_{N-1},n_N} = \lambda_N I_{a^{[1,N]}_{n_1,\ldots,n_N}}$$

for $n_N = 1, 2, \ldots, C_N - 1$, and the lower diagonal blocks are

$$(LD)^{[N]}_{n_1,\ldots,n_{N-1},n_N} = \left[\begin{array}{cc} \alpha_N(1-\delta_{n_N,C_N}p^*_{N,0})I_{s_1} & \underline{0}'_{s_1}\underline{0}_{a^{[1,N-1]}_{n_1,\ldots,n_N}+s_N-s_1} \\ \underline{0}'_{a^{[1,N-1]}_{n_1,\ldots,n_N}+s_N-s_1}\underline{0}_{s_1} & \alpha_N(1-\delta_{n_N,C_N}p^*_{N,0})I_{a^{[1,N-1]}_{n_1,\ldots,n_N}+s_N-s_1} \\ (I_{b_N}-\delta_{n_N,C_N}\mathrm{diag}(\underline{p}^*_N))\underline{B}'_{0,N}\underline{\gamma}_1 & \underline{0}'_{b_N}\underline{0}_{a^{[1,N-1]}_{n_1,\ldots,n_N}+s_N-s_1} \end{array}\right]$$

for $n_N = 1$ and

$$(LD)^{[N]}_{n_1,\ldots,n_{N-1},n_N} = \left[\begin{array}{cc} n_N\alpha_N(1-\delta_{n_N,C_N}p^*_{N,0})I_{a^{[1,N-1]}_{n_1,\ldots,n_N}+s_N} & \underline{0}'_{a^{[1,N-1]}_{n_1,\ldots,n_N}+s_N}\underline{0}_{b_N} \\ \underline{0}'_{b_N}\underline{0}_{a^{[1,N-1]}_{n_1,\ldots,n_N}+s_N} & (I_{b_N}-\delta_{n_N,c_N}\mathrm{diag}(\underline{p}^*_N))((n_N-1)\alpha_N I_{b_N}+\underline{B}'_{0,N}\underline{\beta}_N) \end{array}\right]$$

for $n_N = 2, 3, \ldots, C_N$.

We now build the QBD structured blocks that are needed to track changes in $X_j(t)$, $j = 2, 3, \ldots, N-1$. For $n_i = 0, 1, \ldots, C_i$, $i = 1, 2, \ldots, j-1$, we define

$$Q^{[j]}_{n_1,\ldots,n_{j-1}} = \begin{array}{c|cccccc|} & 0 & 1 & 2 & \cdots & C_j-1 & C_j \\ 0 & Q^{[j+1]}_{n_1,\ldots,n_{j-1},0} & (UD)^{[j]}_{n_1,\ldots,n_{j-1},0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ 1 & (LD)^{[j]}_{n_1,\ldots,n_{j-1},1} & Q^{[j+1]}_{n_1,\ldots,n_{j-1},1} & (UD)^{[j]}_{n_1,\ldots,n_{j-1},1} & \ddots & \mathbf{0} & \mathbf{0} \\ 2 & \mathbf{0} & (LD)^{[j]}_{n_1,\ldots,n_{j-1},2} & Q^{[j+1]}_{n_1,\ldots,n_{j-1},2} & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ C_j-1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q^{[j+1]}_{n_1,\ldots,n_{j-1},c_j-1} & (UD)^{[j]}_{n_1,\ldots,n_{j-1},c_j-1} \\ C_j & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & (LD)^{[j]}_{n_1,\ldots,n_{j-1},c_j} & Q^{[j+1]}_{n_1,\ldots,n_{j-1},c_j} \end{array}.$$

Note how the main diagonal blocks of $Q^{[j]}_{n_1,\ldots,n_{j-1}}$ are simply $Q^{[j+1]}_{n_1,\ldots,n_{j-1},n_j}$, implying that these must be constructed recursively, starting with our original $Q^{[N]}_{n_1,\ldots,n_{N-1}}$ blocks. The upper and lower diagonal blocks make use of a similar recursion in their definitions. The upper diagonal blocks are $(UD)^{[j]}_{n_1,\ldots,n_{j-1},n_j}$, where

$$(UD)^{[j]}_{n_1,\ldots,n_{j+k-1},n_{j+k}} = \begin{bmatrix} (UD)^{[j]}_{n_1,\ldots,n_{j+k},0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & (UD)^{[j]}_{n_1,\ldots,n_{j+k},1} & \ddots & \mathbf{0} \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & (UD)^{[j]}_{n_1,\ldots,n_{j+k},C_{j+k+1}} \end{bmatrix} \tag{5.46}$$

for $k = 0, 1, \ldots, N-j-1$, with

$$(UD)^{[j]}_{n_1,\ldots,n_{N-1},n_N} = \left[\begin{array}{ccc} \lambda_j I_{a^{[1,j-1]}_{n_1,\ldots,n_N}+s_j} & \underline{0}'_{a^{[1,j-1]}_{n_1,\ldots,n_N}+s_j}\underline{0}_{b_j} & \underline{0}'_{a^{[1,j-1]}_{n_1,\ldots,n_N}+s_j}\underline{0}_{a^{[j+1,N]}_{n_1,\ldots,n_N}} \\ \underline{0}'_{a^{[j+1,N]}_{n_1,\ldots,n_N}}\underline{0}_{a^{[1,j-1]}_{n_1,\ldots,n_N}+s_j} & \underline{0}'_{a^{[j+1,N]}_{n_1,\ldots,n_N}}\underline{0}_{b_j} & \lambda_j I_{a^{[j+1,N]}_{n_1,\ldots,n_N}} \end{array}\right]$$

for $n_j = 0$ and

$$(UD)^{[j]}_{n_1,\ldots,n_{N-1},n_N} = \lambda_j I_{a^{[1,N]}_{n_1,\ldots,n_N}}$$

for $n_j = 1, 2, \ldots, C_j - 1$. Similarly, the lower diagonal blocks are $(LD)^{[j]}_{n_1,\ldots,n_{j-1},n_j}$, where

$$(LD)^{[j]}_{n_1,\ldots,n_{j+k-1},n_{j+k}} = \begin{bmatrix} (LD)^{[j]}_{n_1,\ldots,n_{j+k},0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & (LD)^{[j]}_{n_1,\ldots,n_{j+k},1} & \ddots & \mathbf{0} \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & (LD)^{[j]}_{n_1,\ldots,n_{j+k},C_{j+k+1}} \end{bmatrix} \tag{5.47}$$

for $k = 0, 1, \ldots, N - j - 1$, with

$$(LD)^{[j]}_{n_1,\ldots,n_{N-1},n_N} = \begin{bmatrix} \alpha_j(1-\delta_{n_j,C_j}p^*_{j,0})I_{a^{[1,j-1]}_{n_1,\ldots,n_N}+s_j} & \underline{0}'_{a^{[1,j-1]}_{n_1,\ldots,n_N}+s_j}\underline{0}_{s_{j+1}} & \underline{0}'_{a^{[1,j-1]}_{n_1,\ldots,n_N}+s_j}\underline{0}_{a^{[j+1,N]}_{n_1,\ldots,n_N}-s_{j+1}} \\ \underline{0}'_{b_j}\underline{0}_{a^{[1,j-1]}_{n_1,\ldots,n_N}+s_j} & (I_{b_j}-\delta_{n_j,C_j}\mathrm{diag}(\underline{p}^*_j))\underline{B}'_{0,j}\underline{\gamma}_{j+1} & \underline{0}'_{b_j}\underline{0}_{a^{[j+1,N]}_{n_1,\ldots,n_N}-s_{j+1}} \\ \underline{0}'_{s_{j+1}}\underline{0}_{a^{[1,j-1]}_{n_1,\ldots,n_N}+s_j} & \alpha_j(1-\delta_{n_j,C_j}p^*_{j,0})I_{s_{j+1}} & \underline{0}'_{s_{j+1}}\underline{0}_{a^{[j+1,N]}_{n_1,\ldots,n_N}-s_{j+1}} \\ \underline{0}'_{a^{[j+1,N]}_{n_1,\ldots,n_N}-s_{j+1}}\underline{0}_{a^{[1,j-1]}_{n_1,\ldots,n_N}+s_j} & \underline{0}'_{a^{[j+1,N]}_{n_1,\ldots,n_N}-s_{j+1}}\underline{0}_{s_{j+1}} & \alpha_j(1-\delta_{n_j,C_j}p^*_{j,0})I_{a^{[j+1,N]}_{n_1,\ldots,n_N}-s_{j+1}} \end{bmatrix}$$

for $n_j = 1$ and

$$(LD)^{[j]}_{n_1,\ldots,n_{N-1},n_N} =$$

$$\begin{bmatrix} n_j\alpha_j(1-\delta_{n_j,C_j}p^*_{j,0})I_{a^{[1,j-1]}_{n_1,\ldots,n_N}+s_j} & \underline{0}'_{a^{[1,j-1]}_{n_1,\ldots,n_N}+s_j}\underline{0}_{b_j} & \underline{0}'_{a^{[1,j-1]}_{n_1,\ldots,n_N}+s_j}\underline{0}_{a^{[j+1,N]}_{n_1,\ldots,n_N}} \\ \underline{0}'_{b_j}\underline{0}_{a^{[1,j-1]}_{n_1,\ldots,n_N}+s_j} & (I_{b_j}-\delta_{n_j,C_j}\mathrm{diag}(\underline{p}^*_j))((n_j-1)\alpha_j I_{b_j}+\underline{B}'_{0,j}\underline{\beta}_j) & \underline{0}'_{b_j}\underline{0}_{a^{[j+1,N]}_{n_1,\ldots,n_N}} \\ \underline{0}'_{a^{[j+1,N]}_{n_1,\ldots,n_N}}\underline{0}_{a^{[1,j-1]}_{n_1,\ldots,n_N}+s_j} & \underline{0}'_{a^{[j+1,N]}_{n_1,\ldots,n_N}}\underline{0}_{b_j} & n_j\alpha_j(1-\delta_{n_j,C_j}p^*_{j,0})I_{a^{[j+1,N]}_{n_1,\ldots,n_N}} \end{bmatrix}$$

for $n_j = 2, 3, \ldots, C_j$.

Finally, the complete infinitesimal generator is simply the QBD modelling changes in $X_1(t)$, namely

$$Q = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C_1-1 \\ C_1 \end{array} \begin{array}{c} \begin{array}{cccccc} 0 & 1 & 2 & \cdots & C_1-1 & C_1 \end{array} \\ \begin{bmatrix} Q^{[2]}_0 & (UD)^{[1]}_0 & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ (LD)^{[1]}_1 & Q^{[2]}_1 & (UD)^{[1]}_1 & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (LD)^{[1]}_2 & Q^{[2]}_2 & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q^{[2]}_{C_1-1} & (UD)^{[1]}_{C_1-1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & (LD)^{[1]}_{C_1} & Q^{[2]}_{C_1} \end{bmatrix} \end{array},$$

where we again use Equations (5.46) and (5.47), with

$$(UD)^{[1]}_{n_1,\ldots,n_{N-1},n_N} = \begin{bmatrix} \lambda_1 I_{s_1} & \underline{0}'_{s_1}\underline{0}_{b_1} & \underline{0}'_{s_1}\underline{0}_{a^{[2,N]}_{n_1,\ldots,n_N}} \\ \underline{0}'_{a^{[2,N]}_{n_1,\ldots,n_N}}\underline{0}_{s_1} & \underline{0}'_{a^{[2,N]}_{n_1,\ldots,n_N}}\underline{0}_{b_1} & \lambda_1 I_{a^{[2,N]}_{n_1,\ldots,n_N}} \end{bmatrix}$$

for $n_1 = 0$ and

$$(UD)^{[1]}_{n_1,\ldots,n_{N-1},n_N} = \lambda_1 I_{a^{[1,N]}_{n_1,\ldots,n_N}}$$

for $n_1 = 1, 2, \ldots, C_1 - 1$, and

$$(LD)^{[1]}_{n_1,\ldots,n_{N-1},n_N} = \begin{bmatrix} \alpha_1(1-\delta_{n_1,C_1} p^*_{1,0}) I_{s_1} & \underline{0}'_{s_1}\underline{0}_{s_2} & \underline{0}'_{s_1}\underline{0}_{a^{[2,N]}_{n_1,\ldots,n_N}-s_2} \\ \underline{0}'_{b_1}\underline{0}_{s_1} & (I_{b_1} - \delta_{n_1,C_1}\text{diag}(\underline{p}^*_1))\underline{B}'_{0,1}\underline{\gamma}_2 & \underline{0}'_{b_1}\underline{0}_{a^{[2,N]}_{n_1,\ldots,n_N}-s_2} \\ \underline{0}'_{s_2}\underline{0}_{s_1} & \alpha_1(1-\delta_{n_1,C_1} p^*_{1,0}) I_{s_2} & \underline{0}'_{s_2}\underline{0}_{a^{[2,N]}_{n_1,\ldots,n_N}-s_2} \\ \underline{0}'_{a^{[2,N]}_{n_1,\ldots,n_N}-s_2}\underline{0}_{s_1} & \underline{0}'_{a^{[2,N]}_{n_1,\ldots,n_N}-s_2}\underline{0}_{s_2} & \alpha_1(1-\delta_{n_1,C_1} p^*_{1,0}) I_{a^{[2,N]}_{n_1,\ldots,n_N}-s_2} \end{bmatrix}$$

for $n_1 = 1$ and

$$(LD)^{[1]}_{n_1,\ldots,n_{N-1},n_N} =$$

$$\begin{bmatrix} n_1\alpha_1(1-\delta_{n_1,C_1} p^*_{1,0}) I_{s_1} & \underline{0}'_{s_1}\underline{0}_{b_1} & \underline{0}'_{s_1}\underline{0}_{a^{[2,N]}_{n_1,\ldots,n_N}} \\ \underline{0}'_{b_1}\underline{0}_{s_1} & (I_{b_1} - \delta_{n_1,C_1}\text{diag}(\underline{p}^*_1))((n_1-1)\alpha_1 I_{b_1} + \underline{B}'_{0,1}\underline{\beta}_1) & \underline{0}'_{b_1}\underline{0}_{a^{[2,N]}_{n_1,\ldots,n_N}} \\ \underline{0}'_{a^{[2,N]}_{n_1,\ldots,n_N}}\underline{0}_{s_1} & \underline{0}'_{a^{[2,N]}_{n_1,\ldots,n_N}}\underline{0}_{b_1} & n_1\alpha_1(1-\delta_{n_1,C_1} p^*_{1,0}) I_{a^{[2,N]}_{n_1,\ldots,n_N}} \end{bmatrix}$$

for $n_1 = 2, 3, \ldots, C_1$.

### 5.7.4 Model with Patient Customers

If we suppose there is at least one class of patient customers (i.e., $\alpha_i = 0$ for at least one $i = 1, 2, \ldots, N$), then without loss of generality we can label this class as class 1 and shift the indices of all other queues to maintain the same cyclic polling order. By selecting the queue length of this class to represent the level of the process, the QBD in Section 5.7.3 becomes level independent. This allows us to set $C_1 = \infty$ and have an infinite buffer for just that class, resulting in a vector of steady-state probabilities $\underline{\pi} = (\underline{\pi}_0, \underline{\pi}_1, \ldots)$ where the $\underline{\pi}_i$'s are as previously defined.

Suppose now that we have one or more classes with patient customers other than class 1. When a class has patient customers, this should impact the choice of UWC probabilities. Firstly, as departures from such a class are impossible to observe when the server is not at their queue, the value of $p^*_{i,0}$ does not matter so we simply set $p^*_{i,0} = 0$. For when the server is at $Q_i$, we again consider two different versions of UWC. The first version simply lets $\alpha_i = 0$ in Equation (5.43), which will provide a safe choice of $p^*_{i,j}$, $j = 1, 2, \ldots, b_i$, but it may not be optimal. As we observed in Section 5.3, this simplifies to

$$p^*_{i,j} = 1 - \left(\frac{1}{1 - \lambda_i / B_{j,0,i}}\right)^{-1} = \frac{\lambda_i}{B_{j,0,i}},\ j = 1, 2, \ldots, b_i.$$

Note that this is only defined for $j$ where $B_{j,0,i} > 0$ (i.e., when it is possible to observe a service completion from phase $j$). If $B_{j,0,i} = 0$, we can simply let $p^*_{i,j} = 1$ by convention, as it will be impossible to observe class-$i$ customer departures (and hence, queue length decrements) from states where this UWC probability is relevant.

For our second version of UWC for patient customers, we aim to approximate the UWC probability from Section 5.5 where we remarked that

$$\underline{\beta}^*_C = \frac{\underline{\pi}_{C-1}}{\underline{\pi}_{C-1}\underline{e}'} = \frac{\underline{\pi}^{\text{FB}}_{C-1}}{\underline{\pi}^{\text{FB}}_{C-1}\underline{e}'}.$$

As in Section 5.7.1, we use the FB model steady-state probabilities to calculate $\hat{\underline{\beta}}_i^*$ by replacing the IB model probabilities in Equation (5.44). Applying Equation (5.36), our second version of UWC probabilities for class $i$ when $\underline{\alpha}_i = 0$ is

$$p_{i,j}^* = \frac{-\lambda_i \hat{\underline{\beta}}_i^* B_i^{-1} \underline{e}'}{1 - \lambda_i (\hat{\underline{\beta}}_i^* - \underline{\beta}_i) B_i^{-1} \underline{e}'}, \; j = 1, 2, \ldots, b_i.$$

Note that since we are letting $C_1 = \infty$, we do not require UWC probabilities $p_{1,j}^*$. Additionally, if a class $i$ of patient customers require exponentially distributed services, then $\underline{\beta}_i^* = \hat{\underline{\beta}}_i^* = \underline{\beta}_i = 1$ and both versions will result in the same $p_{i,1}^* = \lambda / B_{1,0,i}$ probability. Correspondingly, if every patient class required exponential service, then the two versions of UWC result in identical probabilities and there is no need to undergo the additional computations necessary to analyze the FB model, as we may simply apply the first version.

Making use of either version of UWC probabilities as well as generator blocks previously constructed in Section 5.7.3, the infinitesimal generator for this CTMC is

$$Q = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \\ \vdots \end{array} \begin{array}{c} \begin{array}{ccccc} 0 & 1 & 2 & 3 & \cdots \end{array} \\ \left[\begin{array}{ccccc} Q_0^{[2]} & (UD)_0^{[1]} & \mathbf{0} & \mathbf{0} & \cdots \\ (LD)_1^{[1]} & Q_1^{[2]} & (UD)_1^{[1]} & \mathbf{0} & \cdots \\ \mathbf{0} & (LD)_2^{[1]} & Q_1^{[2]} & (UD)_1^{[1]} & \ddots \\ \mathbf{0} & \mathbf{0} & (LD)_2^{[1]} & Q_1^{[2]} & \ddots \\ \vdots & \vdots & \ddots & \ddots & \ddots \end{array}\right] \end{array}.$$

This results in the system of matrix equations

$$\underline{0} = \underline{\pi}_0 Q_0^{[2]} + \underline{\pi}_1 (LD)_1^{[1]}, \tag{5.48}$$

$$\underline{0} = \underline{\pi}_0 (UD)_0^{[1]} + \underline{\pi}_1 Q_1^{[2]} + \underline{\pi}_2 (LD)_2^{[1]}, \tag{5.49}$$

$$\underline{0} = \underline{\pi}_i (UD)_1^{[1]} + \underline{\pi}_{i+1} Q_1^{[2]} + \underline{\pi}_{i+2} (LD)_2^{[1]}, \; i \in \mathbb{Z}^+. \tag{5.50}$$

As this is a level independent QBD, we have a solution satisfying the matrix geometric form, $\underline{\pi}_i = \underline{\pi}_1 R^{i-1}$, $i \in \mathbb{Z}^+$, and we may apply the algorithm covered in Section 1.2.4 to solve for $R$ followed by the steady-state probabilities.

## 5.8 Numerical Examples

### 5.8.1 Comparing Choices of UWC Probabilities for Patient Customers

To compare the two versions of UWC for patient customers in a $N$-queue polling system, we investigate the following example. We let the polling system have $N = 4$ queues with $\alpha_i = 0$ for $i = 1, 2, 3, 4$. The following three possible pairs of service time distributions are considered, where each pair contains one distribution with mean time 1 and one distribution with mean 2:

- (Exp) Exponential:
  - E[$Ser$] = 1 and Var($Ser$) = 1:

$$Ser \sim \text{PH}_1(\underline{\beta} = 1, B = -1).$$

- $\mathrm{E}[Ser] = 2$ and $\mathrm{Var}(Ser) = 4$:

$$Ser \sim \mathrm{PH}_1(\underline{\beta} = 1, B = -1/2).$$

- ($\mathrm{H}_2$) Hyperexponential-2:
  - $\mathrm{E}[Ser] = 1$ and $\mathrm{Var}(Ser) = 2$:

$$Ser \sim \mathrm{PH}_2 \left( \underline{\beta} = (0.5, 0.5), B = \left[ \begin{array}{cc} -(2+\sqrt{2}) & 0 \\ 0 & -(2-\sqrt{2}) \end{array} \right] \right).$$

  - $\mathrm{E}[Ser] = 2$ and $\mathrm{Var}(Ser) = 8$:

$$Ser \sim \mathrm{PH}_2 \left( \underline{\beta} = (0.5, 0.5), B = \left[ \begin{array}{cc} -(1+\sqrt{1/2}) & 0 \\ 0 & -(1-\sqrt{1/2}) \end{array} \right] \right).$$

- ($\mathrm{E}_2^f$) Erlang-2 with feedback:
  - $\mathrm{E}[Ser] = 1$ and $\mathrm{Var}(Ser) = 0.75$:

$$Ser \sim \mathrm{PH}_2 \left( \underline{\beta} = (1, 0), B = \left[ \begin{array}{cc} -4 & 4 \\ 2 & -4 \end{array} \right] \right).$$

  - $\mathrm{E}[Ser] = 2$ and $\mathrm{Var}(Ser) = 3$:

$$Ser \sim \mathrm{PH}_2 \left( \underline{\beta} = (1, 0), B = \left[ \begin{array}{cc} -2 & 2 \\ 1 & -2 \end{array} \right] \right).$$

Note that the ratios of means and variances are the same for each pair of distributions. Queues 1 and 3 (2 and 4) are assigned the service time distributions with the smaller (larger) means.

We set the arrival rates to be $\lambda_1 = \lambda_4 = 8/45$ and $\lambda_2 = \lambda_3 = 4/45$, which results in a workload of

$$\rho = \sum_{i=1}^{4} \rho_i = \sum_{i=1}^{4} \lambda_i \mathrm{E}[Ser_i] = 0.8.$$

We selected this ordering for the combination of service time distributions and arrival rates as it results in class 1 having the longest expected queue length, which is assigned the infinite buffer $C_1 = \infty$. For simplicity, we let all switch-in times be iid Exp(1).

For the above three cases of service time distributions, we let $C_2 = C_3 = C_4 = C$, and consider $C = 2, 3, \ldots, 10$. The steady-state probabilities are calculated using both versions of UWC as well as for the FB model, and $\mathrm{E}[X_i]$, $i = 1, 2, 3, 4$, is plotted against $C$ in Figure 5.2. In each plot, a light grey horizontal line is included at the corresponding expected queue length from the IB model, which were calculated using the results in Boon [15], Section 2.2.6.

In all cases, the use of either version of UWC provides a benefit over FB, which shrinks as the expected queue lengths approach their limits (i.e., $\mathrm{E}[X_i^{\mathrm{IB}}]$). Under exponential service, the two versions of UWC are identical, as previously discussed. Although not exactly the same under $\mathrm{H}_2$ service, there is minimal difference between them, and so version 1 should be selected as it does not require the increased computation time. However, version 2 greatly outperforms version 1 under $\mathrm{E}_2^f$ service.

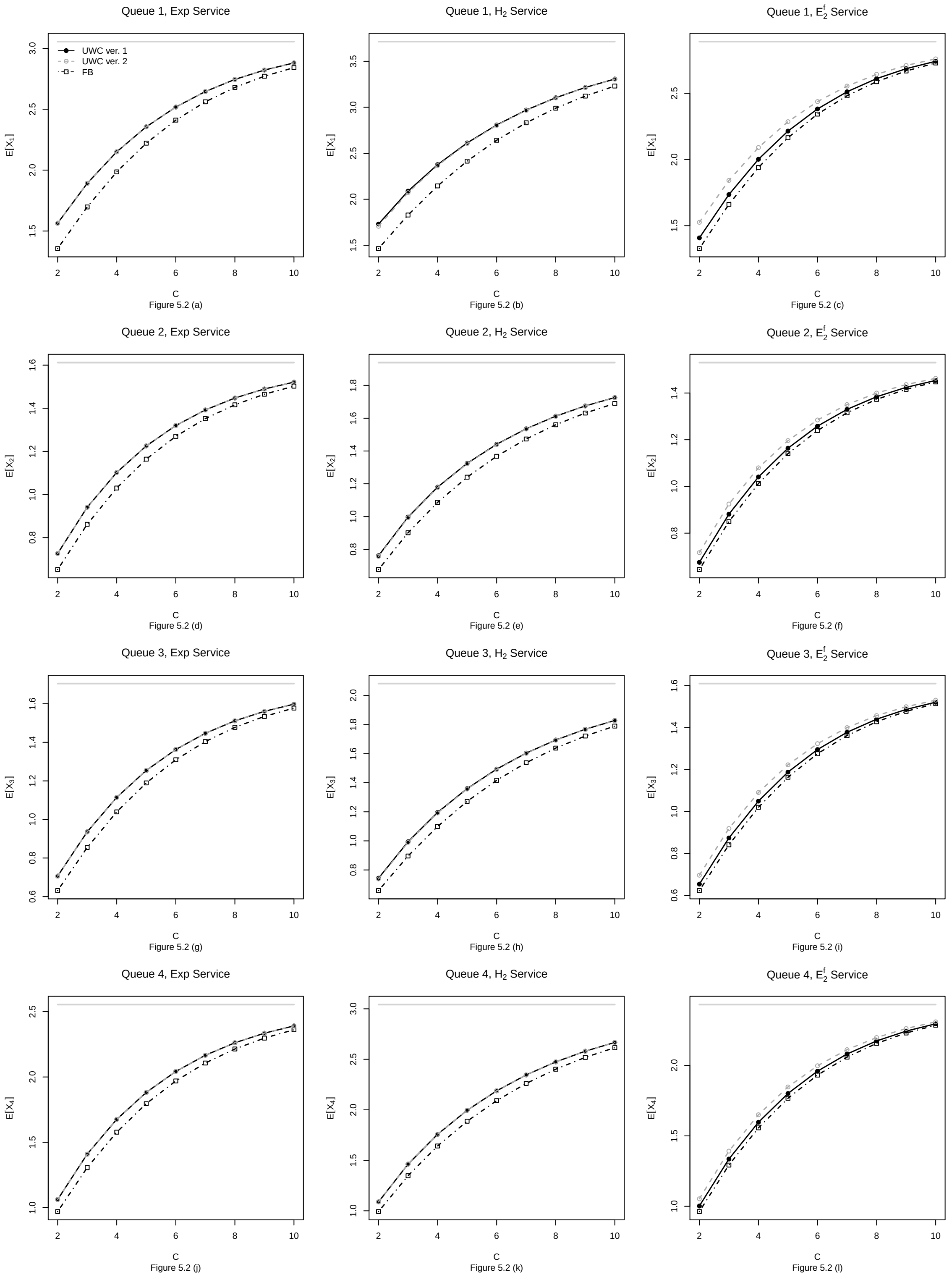

Figure 5.2 (a)
Figure 5.2 (b)
Figure 5.2 (c)
Figure 5.2 (d)
Figure 5.2 (e)
Figure 5.2 (f)
Figure 5.2 (g)
Figure 5.2 (h)
Figure 5.2 (i)
Figure 5.2 (j)
Figure 5.2 (k)
Figure 5.2 (l)

Figure 5.2: Plots of expected marginal queue lengths at steady state in a 4-queue system versus buffers $C_2 = C_3 = C_4 = C$ for UWC version 1, UWC version 2, and FB models, under Exp, $H_2$, or $E_2^f$ service.

In Section 5.6, we previously discussed why the UWC probabilities that we are calling version 1 underperform for distributions that involve service phase transitions that do not result in observed customer departures. As the process of customer departures is more similar between a $M/PH/1+M$ model and a $M/M/1+M$ model than it is between a $M/PH/1$ model and a $M/M/1$ model (i.e., after we remove the reneging when the phase-type service time distribution behaves sufficiently different than an exponential distribution), it is not surprising to observe small gains here in the $\mathrm{E}_2^f$ case for version 2 which is based on what was optimal for a $M/PH/1$ queue.

UWC probabilities are only checked when observing a service completion out of phase 2, and those probabilities are identical to what would have been optimal for exponential distributions possessing means that are only half as large. These correspond to shorter level-$C$ busy periods and fewer required observed customer departures to successfully reduce the queue length. Therefore, less time is spent before reducing a queue length below its buffer and less steady-state probability mass is being shifted. In contrast, the $\mathrm{H}_2$ distributions are similar enough to exponential distributions, in that possible service phase transitions are only observed after service completions, resulting in an acceptable performance by version 1.

As version 2 is based on the UWC probabilities that are optimal in a $M/PH/1$ queue, we would not expect a negative impact on their performance due to a specific phase-type structure. We observe this in Figure 5.2, with the version 2 expected queue lengths receiving comparable gains in accuracy for all considered distributions. Thus, when applying this in practice, it would be safest to go with version 2, however if the corresponding phase-type distributions are exponential or similar to these $\mathrm{H}_2$ distributions, version 1 may be applied, granting similar gains in accuracy without the time cost of additional computations (relative to standard analysis using a FB model).

### 5.8.2 The Impact of Service Phase Transitions in the Presence of Reneging

Continuing from our earlier discussion of the potential impact on the effectiveness of UWC by the switching of service phases without observing customer departures, we compare mean queue lengths between our two UWC models as well as the FB model in a 2-queue system with arrival rates $\lambda_1 = \lambda_2 = 8/15$, reneging rates $\alpha_1 = \alpha_2 = 0.05$, and iid Exp(1) switch-in times. To easily scale the number of service phases while controlling for the expected values, we select Erlang-$k$ ($\mathrm{E}_k$) service time distributions with means of 1 (for class 1) and 2 (for class 2). Note that when there is only one service phase (i.e., $k = 1$), these are simply Exp(1) and Exp(1/2) distributions, respectively.

In Figure 5.3, we plot $\mathrm{E}[X_i]$ for both classes for the corresponding UWC and FB models. As an impatient customer example, we treat this as a level-dependent QBD and let $C_1 = C_2 = C$, $C = 2, 3, \ldots, 20$. The number of service phases are similarly kept constant between the classes, and we present the cases for $k = 1, 2, 5, 10$. Within these plots, the light grey horizontal lines are approximated $\mathrm{E}[X_i^{\mathrm{IB}}]$ values, obtained via calculating the mean queue lengths for the FB model with $C = 40$. As we would expect, we would rank UWC version 2 above version 1, with both outperforming the FB model in all cases. When $k = 1$, both UWC versions are identical due to the presence of only a single service phase. Performance is comparable for UWC version 2 across all cases, while version 1 has the widest margins between itself and the FB model in the $k = 1$ case, which we know enables the best performance by this version of UWC. We also observe in all cases that the difference in effectiveness between the UWC versions decrease

Table 5.3: UWC version 1 and UWC version 2 model steady-state expected marginal queue length convergence percentages ($\mathrm{E}[X_i^{\mathrm{UWC}}]/\mathrm{E}[X_i^{\mathrm{IB}}]$, $i = 1, 2$) at various buffers $C_1 = C_2 = C$ under $\mathrm{E}_k$ service, $k = 1, 2, 5, 10$.

| Class 1, UWC ver. 1 | | | | | $C$ | | | | |
|---|---|---|---|---|---|---|---|---|---|
| $k$ | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 1 | 0.2788 | 0.4036 | 0.5172 | 0.6193 | 0.7090 | 0.7852 | 0.8473 | 0.8957 | 0.9315 |
| 2 | 0.2464 | 0.3693 | 0.4864 | 0.5944 | 0.6905 | 0.7725 | 0.8391 | 0.8906 | 0.9285 |
| 5 | 0.2294 | 0.3532 | 0.4728 | 0.5835 | 0.6822 | 0.7664 | 0.8348 | 0.8877 | 0.9266 |
| 10 | 0.2245 | 0.3490 | 0.4694 | 0.5808 | 0.6800 | 0.7647 | 0.8335 | 0.8868 | 0.9260 |

| Class 1, UWC ver. 2 | | | | | $C$ | | | | |
|---|---|---|---|---|---|---|---|---|---|
| $k$ | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 1 | 0.2788 | 0.4036 | 0.5172 | 0.6193 | 0.7090 | 0.7852 | 0.8473 | 0.8957 | 0.9315 |
| 2 | 0.2762 | 0.3998 | 0.5129 | 0.6150 | 0.7050 | 0.7819 | 0.8447 | 0.8937 | 0.9302 |
| 5 | 0.2740 | 0.3967 | 0.5094 | 0.6115 | 0.7018 | 0.7790 | 0.8424 | 0.8920 | 0.9289 |
| 10 | 0.2730 | 0.3955 | 0.5080 | 0.6101 | 0.7004 | 0.7778 | 0.8414 | 0.8912 | 0.9283 |

| Class 2, UWC ver. 1 | | | | | $C$ | | | | |
|---|---|---|---|---|---|---|---|---|---|
| $k$ | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 1 | 0.3810 | 0.5351 | 0.6615 | 0.7616 | 0.8380 | 0.8940 | 0.9333 | 0.9597 | 0.9766 |
| 2 | 0.3478 | 0.4964 | 0.6249 | 0.7317 | 0.8163 | 0.8800 | 0.9251 | 0.9553 | 0.9744 |
| 5 | 0.3283 | 0.4758 | 0.6069 | 0.7176 | 0.8063 | 0.8733 | 0.9211 | 0.9530 | 0.9731 |
| 10 | 0.3220 | 0.4697 | 0.6020 | 0.7140 | 0.8038 | 0.8717 | 0.9200 | 0.9523 | 0.9727 |

| Class 2, UWC ver. 2 | | | | | $C$ | | | | |
|---|---|---|---|---|---|---|---|---|---|
| $k$ | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 1 | 0.3810 | 0.5351 | 0.6615 | 0.7616 | 0.8380 | 0.8940 | 0.9333 | 0.9597 | 0.9766 |
| 2 | 0.3949 | 0.5506 | 0.6762 | 0.7738 | 0.8471 | 0.9002 | 0.9373 | 0.9620 | 0.9779 |
| 5 | 0.4056 | 0.5624 | 0.6869 | 0.7823 | 0.8531 | 0.9041 | 0.9395 | 0.9632 | 0.9784 |
| 10 | 0.4097 | 0.5670 | 0.6910 | 0.7854 | 0.8552 | 0.9053 | 0.9401 | 0.9635 | 0.9785 |

as $C$ is increased, corresponding to customer departures due to reneging making up larger proportions of observed departures during a level-$C$ busy period.

This combination of parameters resulted in values of $\mathrm{E}[X_i^{\mathrm{IB}}]$ between 6.0212 ($k = 1$) and 6.3087 ($k = 10$) for class 1, and 4.3682 ($k = 1$) and 4.1088 ($k = 10$) for class 2. Due to the narrow range of limiting values, this allows us to more accurately compare rates of convergence. In Table 5.3, we present $\mathrm{E}[X_i^{\mathrm{UWC}}]/\mathrm{E}[X_i^{\mathrm{IB}}]$ for $C = 2, 3, \ldots, 10$ for both versions of UWC.

As previously observed, the convergence percentages for the $k = 1$ cases of UWC version 1 are noticeably higher than those for $k \geq 2$. In fact, the difference in percentage for a given $C$ between $k = 1$ and $k = 2$ is larger than that between $k = 2$ and $k = 10$! In contrast, the percentages for UWC version 2 are not very sensitive to changes in $k$, and even increase for class 2 (due to the fact that $\mathrm{E}[X_2^{\mathrm{IB}}]$ is decreasing in $k$). Therefore, in the absence of exponential service time distributions or moderate to large $C$ (alternatively, large reneging rates), UWC version 2 should be used for service time distributions involving multiple phase transitions.

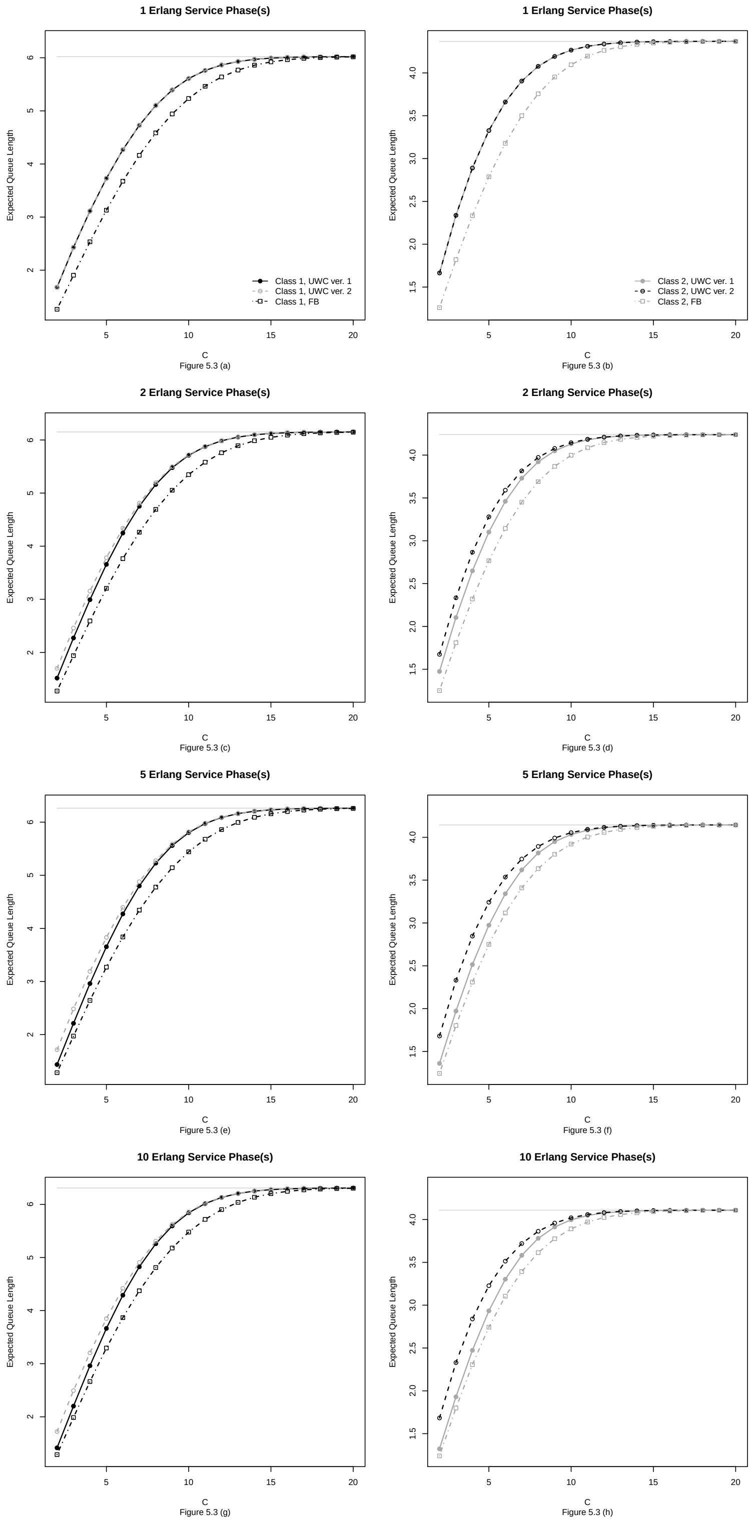

Figure 5.3 (a) Figure 5.3 (b)

Figure 5.3 (c) Figure 5.3 (d)

Figure 5.3 (e) Figure 5.3 (f)

Figure 5.3 (g) Figure 5.3 (h)

Figure 5.3: Plots of expected marginal queue lengths at steady state in a 2-queue system versus buffers $C_1 = C_2 = C$ for UWC version 1, UWC version 2, and FB models, under $\mathrm{E}_k$ service, $k = 1, 2, 5, 10$.

### 5.8.3 Examining Marginal Queue Length Probabilities

In Section 5.6, we considered a numerical example which compared the steady-state probabilities of the UWC version 1 and 2 models of a $M/PH/1+M$ queue against those of the FB model. Since expanding our scope to a $N$-queue system, we have only considered expected queue lengths, so it is of value to examine a similar example. For the benefit of simplified (and condensed) presentation of data, we consider steady-state probabilities for marginal queue lengths (rather than for individual states), and we limit ourselves to a 2-queue system. We allow the service time distributions for classes 1 and 2 to be the $\mathrm{E}_2^f$ distributions considered in Section 5.8.1, with class 1 taking the distribution with the smaller mean. Additionally, as in Section 5.8.2, we let $\lambda_1 = \lambda_2 = 8/15$ and $\alpha_1 = \alpha_2 = 0.05$, while switch-in times are assumed to be iid Exp(1) random variables.

In Figures 5.4 and 5.5, we present barplots of the marginal queue length probabilities for classes 1 and 2, respectively, for both versions of UWC as well as FB models at even buffer sizes $C_1 = C_2 = C$, $C = 2, 4, \ldots, 16$. Plotted along with these values are those from the corresponding IB model, approximated via a FB model with $C = 40$, which are unchanged between plots within a figure.

Unlike the simple single queue case, version 2 does not immediately result in near exact steady-state probabilities for levels below $C$. While there is still some error present as a result of using the FB probabilities, we more importantly do not separate the cases (in terms of UWC probability) where the server begins a level-$C$ busy period due to an arrival versus after a switch-in time. If the busy period begins after a switch, then like in our earlier discussion of the original $M/PH/1+M$ queue UWC version 1 approximation, any cases where possible unobserved customers could be in the system at this instant are treated as if there are exactly zero unobserved customers. Unsurprisingly, this results in a failure to capture all excess probability mass at level $C$. However, for both classes, we are in fact observing the intended effect of probability mass being shifted to the truncation level $C$, resulting in better approximations at lower levels. The relative difference in gains by the two versions over the FB model are larger for small $C$ and decrease as $C$ is increased, consistent with what we have seen previously.

For moderate values of $C$, we again observe instances of underestimating the IB model steady-state probabilities at queue lengths near the truncation level. While more common for UWC, it is also observed for FB (e.g., case $C = 10$ in Figure 5.5). Fortunately, even if the steady-state probabilities are slightly underestimated by UWC, they are still generally closer to the target probabilities than those of the FB model at the same value of $C$, and the underestimation vanishes as $C$ is increased. These results indicate that the either version of the UWC model would indeed be preferable to the FB model at any given $C$. This experiment was also replicated in a more optimal case using exponentially distributed service times (the results of which we omit), which led to the same conclusions while observing higher relative accuracy gains by UWC (with the largest relative gains at small $C$). Overall, we maintain the same conclusion that version 2 is preferable although version 1 is comparable at moderate to high $C$.

### 5.8.4 The Impact of Reneging on Relative Accuracy Gains of UWC Version 1 for Exp or $\mathrm{E}_2^f$ Service Times

Up to now, we have observed that UWC version 1 provides the largest gains in accuracy (in direct comparison to the FB model) when service times are exponentially distributed. The

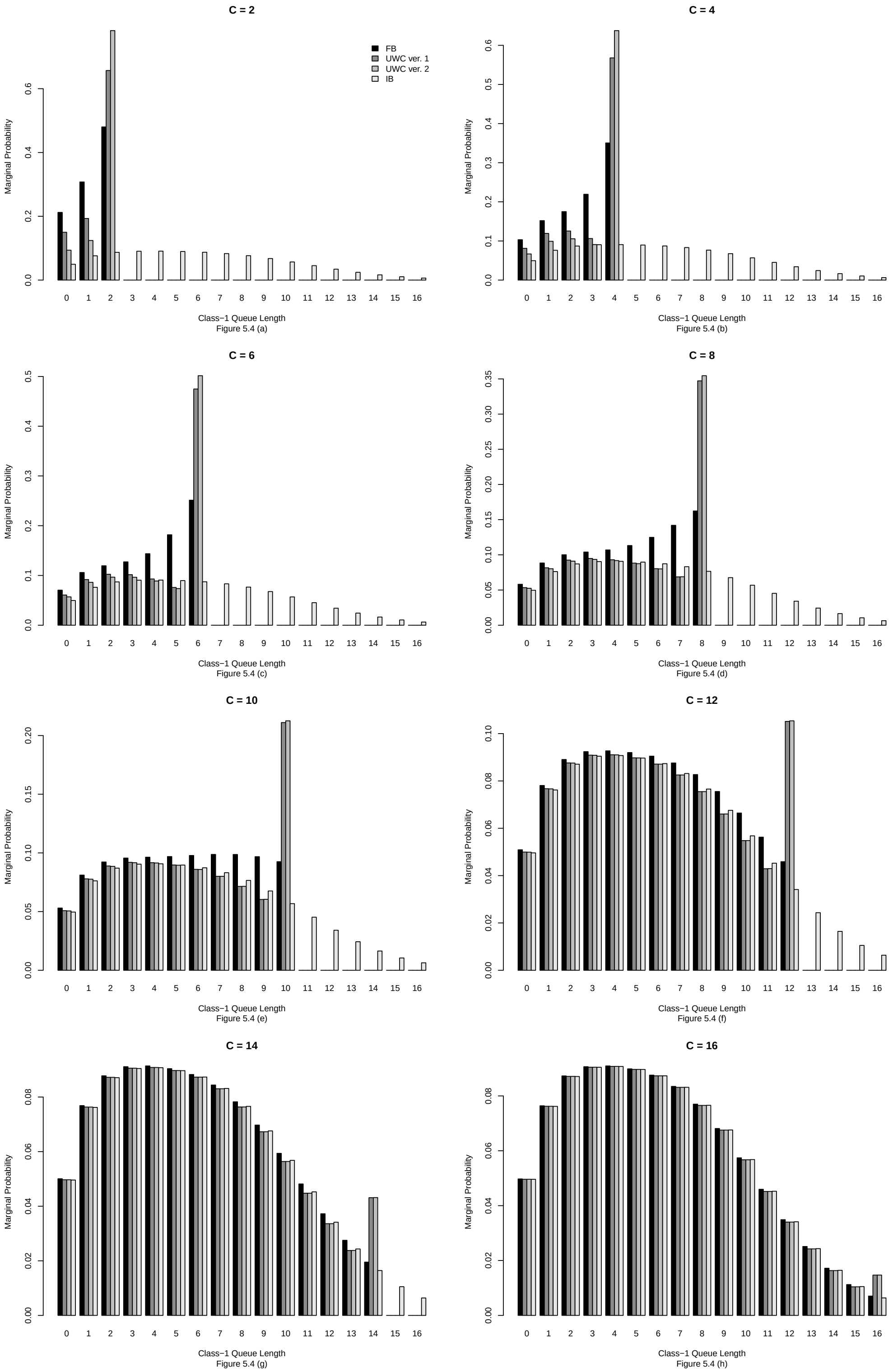

Figure 5.4 (a)

Figure 5.4 (b)

Figure 5.4 (c)

Figure 5.4 (d)

Figure 5.4 (e)

Figure 5.4 (f)

Figure 5.4 (g)

Figure 5.4 (h)

Figure 5.4: Barplots of class-1 marginal queue length probabilities at steady state in a 2-queue system versus buffers $C_1 = C_2 = C$ for UWC version 1, UWC version 2, FB, and IB models, under $\mathrm{E}_2^f$ service.

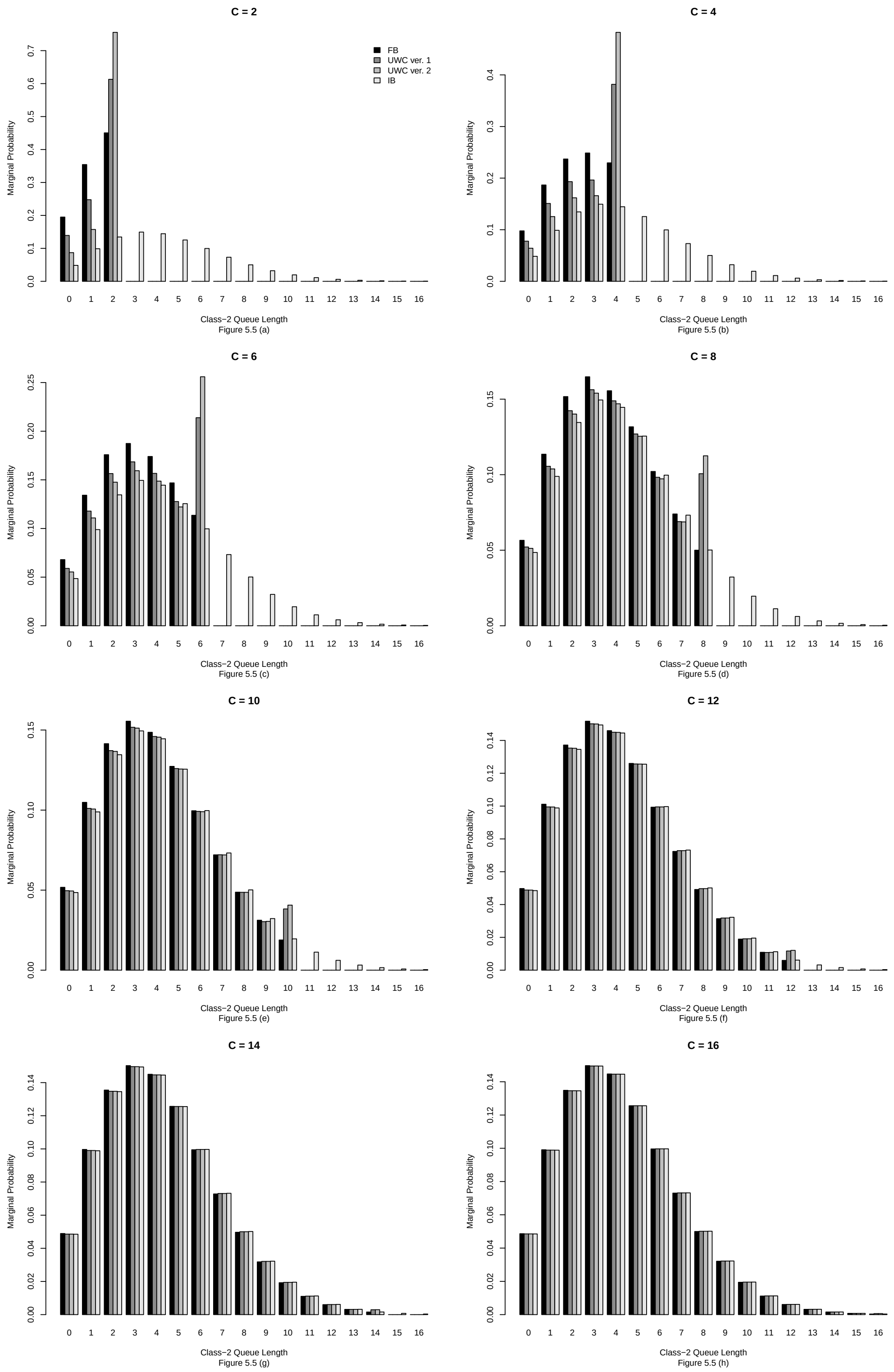

Figure 5.5 (a)

Figure 5.5 (b)

Figure 5.5 (c)

Figure 5.5 (d)

Figure 5.5 (e)

Figure 5.5 (f)

Figure 5.5 (g)

Figure 5.5 (h)

Figure 5.5: Barplots of class-2 marginal queue length probabilities at steady state in a 2-queue system versus buffers $C_1 = C_2 = C$ for UWC version 1, UWC version 2, FB, and IB models, under $\mathrm{E}_2^f$ service.

Table 5.4: Approximated $\mathrm{E}[X_i^{\mathrm{IB}}]$ values for Exp and $\mathrm{E}_2^f$ service.

| | Exp | | | $\mathrm{E}_2^f$ | | |
|---|---|---|---|---|---|---|
| $\alpha$ | $\mathrm{E}[X_1^{\mathrm{IB}}]$ | $\mathrm{E}[X_2^{\mathrm{IB}}]$ | $\mathrm{E}[X_3^{\mathrm{IB}}]$ | $\mathrm{E}[X_1^{\mathrm{IB}}]$ | $\mathrm{E}[X_2^{\mathrm{IB}}]$ | $\mathrm{E}[X_3^{\mathrm{IB}}]$ |
| 0.05 | 5.6104 | 3.0509 | 3.3623 | 5.5874 | 3.0365 | 3.3672 |
| 0.075 | 4.0731 | 2.2651 | 2.3665 | 4.0526 | 2.2533 | 2.3679 |
| 0.1 | 3.2794 | 1.8533 | 1.8623 | 3.2617 | 1.8438 | 1.8624 |

primary cause of the reduction in UWC's effectiveness is service phase transitions to phases with different absorption rates. While this may be a transition not corresponding to a service completion, it is also possible to observe a service completion out of one phase and then initialize the next service time in a different phase (which will be relevant to UWC's effectiveness assuming the observed queue length does not decrement with this departure).

When modelling a system with impatient customers, it is possible that potential queue decrements are triggered by reneging and will not result in a change of service phase. We have seen that as $C$ is increased, the effectiveness of version 1 approaches that of version 2. It should also follow that if customers' reneging rates are larger for a given $C$, then it is likely that we would observe fewer phase transitions in a level-$C$ busy period. It would then follow that the relatively higher gains made by UWC version 1 when service times are exponential (versus $\mathrm{E}_2^f$) should be smaller in a system with higher reneging rates.

To this end, we consider a 3-queue system with $\lambda_1 = 16/25$, $\lambda_2 = \lambda_3 = 8/25$, and $\alpha_1 = \alpha_2 = \alpha_3 = \alpha \in \{0.05, 0.075, 0.1\}$. Service time distributions are either the Exp or $\mathrm{E}_2^f$ distributions used for classes 1, 2, and 3 in Section 5.8.1 (with means of 1, 2, and 1, respectively), while we let all switch-in times be iid Exp(1). Similar to Table 5.3, we will be calculating the convergence percentages of expected marginal queue lengths for both UWC and FB models, while approximating $\mathrm{E}[X_i^{\mathrm{IB}}]$ via a corresponding FB model with $C_1 = C_2 = C_3 = C = 25$. The approximated IB model values are presented in Table 5.4. As the change in service time distribution has minimal effect on these mean queue lengths, this provides an ideal opportunity to directly compare rates of convergence.

Mean queue lengths are calculated for UWC and FB models at $C_1 = C_2 = C_3 = C$, $C = 2, 3, \ldots, 15$, and the mean queue lengths from Table 5.4 are used to approximate the convergence percentages. Our interest is in the difference in accuracy gained by the use of UWC version 1 between the two sets of service time distributions, where for example we let the accuracy gain by the UWC model using Exp service times equal

$$\frac{\mathrm{E}[X_{i,C}^{\mathrm{UWC,Exp}}] - \mathrm{E}[X_{i,C}^{\mathrm{FB,Exp}}]}{\mathrm{E}[X_i^{\mathrm{IB,Exp}}]},$$

where $\mathrm{E}[X_{i,C}^{\mathrm{UWC,Exp}}]$ denotes the steady-state expected marginal class-$i$ queue length of the UWC model having buffer $C$. Considering the difference in these calculated for Exp and $\mathrm{E}_2^f$ service times (for a given class $i$), as $C \to \infty$, both UWC and FB means will converge to the same value, and hence the difference must trend to zero.

We plot the gains in accuracy as well as their differences in Figure 5.6. The gains are concave, increasing up until the buffer surpasses $\mathrm{E}[X_i^{\mathrm{IB}}]$. After this point, the impact of further

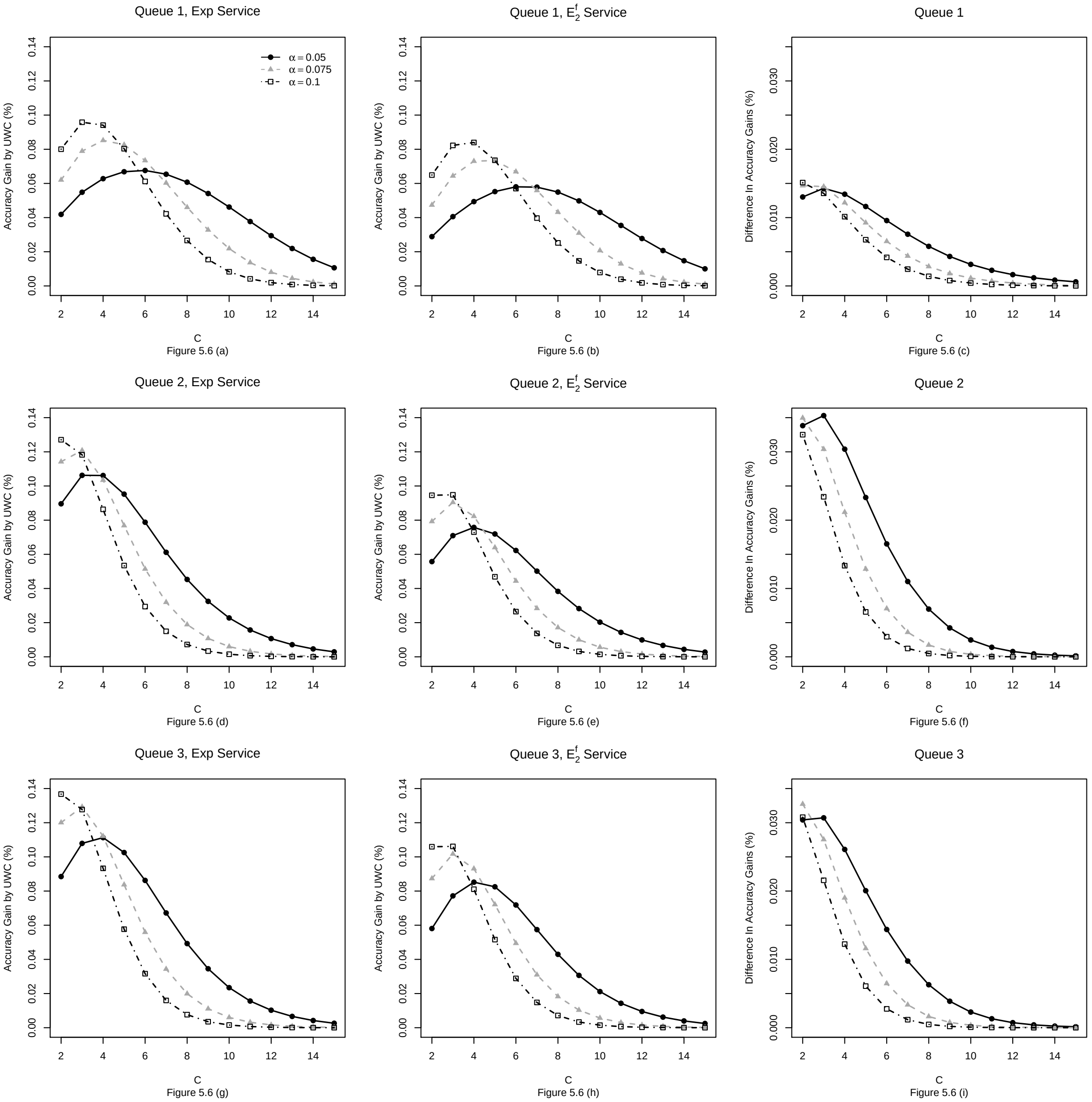

Figure 5.6 (a)
Figure 5.6 (b)
Figure 5.6 (c)
Figure 5.6 (d)
Figure 5.6 (e)
Figure 5.6 (f)
Figure 5.6 (g)
Figure 5.6 (h)
Figure 5.6 (i)

Figure 5.6: Plots of difference in accuracy gained by the use of UWC version 1 in a 3-queue system between models having Exp or $E_2^f$ service versus buffers $C_1 = C_2 = C_3 = C$ for $\alpha_1 = \alpha_2 = \alpha_3 = \alpha \in \{0.05, 0.075, 0.1\}$.

increases in $C$ are lessened as the amount of shifted probability mass becomes less drastic. As $C \to \infty$, the UWC and FB means converge and these gains go to zero (resulting in the difference in gains also going to zero).

For the considered cases, the difference in accuracy gains are no larger than approximately 3.5%. The UWC approximation has larger gains for Exp service at smaller values of $C$ where there is more probability mass at the buffer (implying that the CTMC consults the UWC probabilities more frequently). Additionally, this relative impact is widened when $\mathrm{E}[X_i^{\mathrm{IB}}]$ is smaller, allowing the initial $\mathrm{E}[X_{i,C}^{\mathrm{UWC}}]$ (which must be strictly between 0 and $C$) to be closer to its target value. However, we note that these differences are not solely dependent on the mean IB model marginal queue lengths, as $\mathrm{E}[X_1^{\mathrm{IB}}]$ at $\alpha = 0.1$ is less than $\mathrm{E}[X_3^{\mathrm{IB}}]$ at $\alpha = 0.05$, yet the former has smaller differences.

There is little change in the difference in gains when varying $\alpha$ at $C = 2$, as the total force of reneging at the buffer will still be small relative to the service distribution absorption rates. However, as we increase $C$, the different $\alpha$ cases split, demonstrating smaller differences between the gains of UWC for the two sets of service time distributions for larger reneging rates, confirming our suspicions.

# Chapter 6

# An Application of UWC on a $N$-Class Polling Model with $k_i$-Limited or Bernoulli Service

## 6.1 Model Assumptions

In Chapter 4 we considered a 2-queue system having level-dependent reneging and $k_i$-limited service, while in Chapter 5 we introduced the UWC approximation and applied it to a $N$-queue system with constant class-dependent reneging rates and a simple exhaustive service discipline. We will now simultaneously generalize both models of these preceding chapters and update our UWC version 2 calculations to accommodate level-dependent reneging.

We again consider a cyclic polling system consisting of $N$ queues, $Q_1, Q_2, \ldots, Q_N$, which is attended to by a lone server. Customers within a given queue receive service according to a FCFS policy, and the service discipline for the server at each queue may vary, taking on a class-dependent $k_i$-limited or Bernoulli service policy. Recalling from Section 1.2.7, in the former case, the server will switch away from $Q_i$ after serving $k_i \in \mathbb{Z}^+$ class-$i$ customers, or after the queue empties, whichever happens first. In the latter case, if the server finds a positive queue length at their visitation epoch then they will accept a customer into service. After that service completion (and every other within the same visit to $Q_i$), they will accept another class-$i$ customer into service with probability $p_i^{\mathrm{B}}$ should $Q_i$ not be empty, and depart the queue otherwise. The Bernoulli service discipline reduces to the exhaustive service discipline considered in Chapter 5 if we let $p_i^{\mathrm{B}} = 1$. For convenience in the following model construction, we let $k_i = -1$ if $Q_i$ has Bernoulli (or exhaustive) service.

For $i = 1, 2, \ldots, N$, we assume that class-$i$ service times are iid $\mathrm{PH}_{b_i}(\underline{\beta}_i, B_i)$ random variables while the time for the server to switch into $Q_i$ from $Q_{i-1}$ (letting $Q_0$ represent $Q_N$) are iid $\mathrm{PH}_{s_i}(\underline{\gamma}_i, S_i)$. Additionally, we suppose that all of these times are strictly positive, such that $\underline{\beta}\underline{e}' = \underline{\gamma}\underline{e}' = 1$. We denote the column vectors of absorption rates by $\underline{B}_{0,i}' = -B_i\underline{e}'$ and $\underline{S}_{0,i}' = S_i\underline{e}'$.

Any customer who is not currently receiving service is at risk of reneging from the system due to impatience. Let $\alpha_{i,n}$ denote the current exponential reneging rate of a class-$i$ customer having $n-1$ other class-$i$ customers waiting ahead of them in their queue. Additionally, we define the combined reneging rate of $j$ waiting class-$i$ customers as $\alpha_i^{[j]} = \sum_{n=1}^{j} \alpha_{i,n}$, and by convention let $\alpha_i^{[0]} = 0$. As we are only considering impatient customers, our model will take the form of a level-dependent QBD and we must truncate the queue length of all classes. Denote the finite buffer of $Q_i$ by $C_i < \infty$. The calculation of UWC probabilities $p^*_{i,j}$ will be discussed in Section 6.4 after model construction.

## 6.2 State Space and Steady-State Probabilities

The queueing network is modelled by the CTMC

$$\{(X_1(t), X_2(t), \ldots, X_N(t), L(t), K(t), Y(t)), t \geq 0\},$$

where $X_i(t) \in \{0, 1, \ldots, C_i\}$ is the number of class-$i$ customers, $i = 1, 2, \ldots, N$, $L(t) \in \{1, 2, \ldots, 2N-1, 2N\}$ represents the location of the server, with $L(t) = 2i-1$ if the server is switching into class $i$, or $L(t) = 2i$ if the server is serving a class-$i$ customer, $K(t) \in \{1, \ldots, |k_i|\}$ counts what number service the server is on during a visit to $Q_i$ if using $k_i$-limited service (or simply takes a dummy value of 1 if class $i$ has Bernoulli service) and will be set to 0 during a switch, and $Y(t)$ tracks the phase of the current service or switch-in time. The observable values of $L(t)$ depend on the queue lengths, such that

$$L(t) \in \Omega_L(X_1(t), X_2(t), \ldots, X_N(t)) = \bigcup_{i=1}^{N} \Omega_L(X_i(t)),$$

where

$$\Omega_L(X_i(t)) = \begin{cases} \{2i-1\} & \text{, if } X_i(t) = 0, \\ \{2i-1, 2i\} & \text{, if } X_i(t) > 0. \end{cases}$$

Additionally, the values of $K(t)$ and $Y(t)$ depend on the server's position, with

$$K(t) \in \Omega_K(L(t)) = \begin{cases} \{0\} & \text{, if } L(t) = 1, \\ \{1, \ldots, |k_1|\} & \text{, if } L(t) = 2, \\ & \vdots \\ \{0\} & \text{, if } L(t) = 2i-1, \\ \{1, \ldots, |k_i|\} & \text{, if } L(t) = 2i, \\ & \vdots \\ \{0\} & \text{, if } L(t) = 2N-1, \\ \{1, \ldots, |k_N|\} & \text{, if } L(t) = 2N, \end{cases}$$

and

$$Y(t) \in \Omega_Y(L(t)) = \begin{cases} \{1, 2, \ldots, s_1\} & \text{, if } L(t) = 1, \\ \{1, 2, \ldots, b_1\} & \text{, if } L(t) = 2, \\ & \vdots \\ \{1, 2, \ldots, s_i\} & \text{, if } L(t) = 2i-1, \\ \{1, 2, \ldots, b_i\} & \text{, if } L(t) = 2i, \\ & \vdots \\ \{1, 2, \ldots, s_N\} & \text{, if } L(t) = 2N-1, \\ \{1, 2, \ldots, b_N\} & \text{, if } L(t) = 2N. \end{cases}$$

In contrast to the state space of the $N$-queue exhaustive polling system, the only change is now we must track how many customers of a particular class we have sequentially served (assuming $k_i \geq 1$) *while currently serving that class.* Letting $s = \sum_{i=1}^{N} s_i$, it is clear that we can simply extend Equation (5.45) as follows to obtain the total number of states:

$$\ell = s\prod_{i=1}^{N}(C_i + 1) + \sum_{j=1}^{N} b_j|k_j|\prod_{i=1}^{N}(C_i + 1 - \delta_{i,j}). \tag{6.1}$$

Let $\pi_{n_1,n_2,\ldots,l,k,y}$ be the steady-state probability of the truncated CTMC being in state $(n_1, n_2, \ldots, n_N, l, k, y)$. These probabilities are sorted into ordered row vectors in the following manner. For $i = 1, 2, \ldots, N$, we let

$$\underline{\pi}_{n_1,n_2,\ldots,n_N,l,k} = \begin{cases} (\pi_{n_1,n_2,\ldots,n_N,l,k,1}, \pi_{n_1,n_2,\ldots,n_N,l,k,2}, \ldots, \pi_{n_1,n_2,\ldots,n_N,l,k,s_i}) & \text{, if } l = 2i - 1, \\ (\pi_{n_1,n_2,\ldots,n_N,l,k,1}, \pi_{n_1,n_2,\ldots,n_N,l,k,2}, \ldots, \pi_{n_1,n_2,\ldots,n_N,l,k,b_i}) & \text{, if } l = 2i, \end{cases}$$

from which we obtain

$$\underline{\pi}^{[i]}_{n_1,n_2,\ldots,n_N} = \begin{cases} \underline{\pi}_{n_1,n_2,\ldots,n_N,2i-1,0} & \text{, if } n_i = 0, \\ (\underline{\pi}_{n_1,n_2,\ldots,n_N,2i-1,0}, \underline{\pi}_{n_1,n_2,\ldots,n_N,2i,1}, \ldots, \underline{\pi}_{n_1,n_2,\ldots,n_N,2i,|k_i|}) & \text{, if } n_i > 0, \end{cases}$$

and

$$\underline{\pi}_{n_1,n_2,\ldots,n_N} = (\underline{\pi}^{[1]}_{n_1,n_2,\ldots,n_N}, \underline{\pi}^{[2]}_{n_1,n_2,\ldots,n_N}, \ldots, \underline{\pi}^{[N]}_{n_1,n_2,\ldots,n_N}).$$

Finally, the $C_1+1$ component vectors $\underline{\pi}_{n_1}$ of the combined probability row vector $\underline{\pi} = (\underline{\pi}_0, \underline{\pi}_1, \ldots, \underline{\pi}_{C_1})$ are constructed as

$$\begin{aligned} \underline{\pi}_{n_1} &= (\underline{\pi}_{n_1,0}, \underline{\pi}_{n_1,1}, \ldots, \underline{\pi}_{n_1,C_2}), && 0 \leq n_1 \leq C_1, \\ \underline{\pi}_{n_1,n_2} &= (\underline{\pi}_{n_1,n_2,0}, \underline{\pi}_{n_1,n_2,1}, \ldots, \underline{\pi}_{n_1,n_2,C_3}), && 0 \leq n_2 \leq C_2, \\ &\vdots \\ \underline{\pi}_{n_1,n_2,\ldots,n_i} &= (\underline{\pi}_{n_1,n_2,\ldots,n_i,0}, \underline{\pi}_{n_1,n_2,\ldots,n_i,1}, \ldots, \underline{\pi}_{n_1,n_2,\ldots,n_i,C_{i+1}}), && 0 \leq n_i \leq C_i,\ i = 1, 2, \ldots, N-1. \end{aligned}$$

## 6.3 Infinitesimal Generator Matrix

The recursive structure of our generator matrix construction is analogous to that of Section 5.7.3. First, we require the following definitions:

$$\lambda_{n_1,\ldots,n_N} = \sum_{i=1}^{N} \lambda_i(1 - \delta_{n_i,C_i}),$$

$$a^{[m,n]}_{n_1,\ldots,n_N} = \sum_{i=m}^{n}(s_i + \bar{\delta}_{n_i,0}b_i|k_i|),\ 1 \leq m \leq n \leq N,$$

$$\underline{p}^*_i = (p^*_{i,1}, p^*_{i,2}, \ldots, p^*_{i,b_i}),$$

$$p^*_{i,n_i,l,y} = \begin{cases} p^*_{i,0} & \text{, if } n_i = C_i,\ l \neq 2i, \\ p^*_{i,y} & \text{, if } n_i = C_i,\ l = 2i, \\ 0 & \text{, otherwise,} \end{cases}$$

$$\alpha_{n_1,\ldots,n_N,l,y} = \sum_{i=1}^{N} \alpha_i^{[n_i-\delta_{l,2i}]}(1 - p^*_{i,n_i,l,y}),$$

$$\underline{\alpha}_{n_1,\ldots,n_N,l} = \begin{cases} (\alpha_{n_1,\ldots,n_N,l,1}, \alpha_{n_1,\ldots,n_N,l,2}, \ldots, \alpha_{n_1,\ldots,n_N,l,s_i}) & , \text{ if } l = 2i-1, \\ (\alpha_{n_1,\ldots,n_N,l,1}, \alpha_{n_1,\ldots,n_N,l,2}, \ldots, \alpha_{n_1,\ldots,n_N,l,b_i}) & , \text{ if } l = 2i, \end{cases}$$

and for $j = 1, 2, \ldots, N$,

$$\zeta_{n_1,\ldots,n_N,l} = \begin{cases} S_j - \lambda_{n_1,\ldots,n_N} I_{s_j} - \text{diag}(\underline{\alpha}_{n_1,\ldots,n_N,2j-1}) & , \text{ if } l = 2j-1, \\ B^*_{j,n_j} - I_{|k_j|} \otimes (\lambda_{n_1,\ldots,n_N} I_{b_j} + \text{diag}(\underline{\alpha}_{n_1,\ldots,n_N,2j})) & , \text{ if } l = 2j, \end{cases}$$

and

$$\zeta^*_{n_1,\ldots,n_N,2j} = \begin{cases} \delta_{n_j,C_j}(1 - p_j^{\text{B}})\text{diag}(\underline{p}_j^*)\underline{B}'_{0,j}\underline{\gamma}_{j+1} & , \text{ if } k_j = -1, \\ \delta_{n_j,C_j}\underline{e}'_{k_j,k_j} \otimes \text{diag}(\underline{p}_j^*)\underline{B}'_{0,j}\underline{\gamma}_{j+1} & , \text{ if } k_j \geq 1, \end{cases}$$

where we let $s_{N+1} = s_1$, $\underline{\gamma}_{N+1} = \underline{\gamma}_1$, and

$$B^*_{j,n_j} = \begin{cases} B_j + \delta_{n_j,C_j} p_j^{\text{B}} \text{diag}(\underline{p}_j^*)\underline{B}'_{0,j}\underline{\beta}_j & , \text{ if } k_j = -1, \\ B_j & , \text{ if } k_j = 1, \\ I_{k_j} \otimes B_j + \begin{bmatrix} \underline{0}'_{k_j-1} & I_{k_j-1} \\ 0 & \underline{0}_{k_j-1} \end{bmatrix} \otimes \delta_{n_j,C_j} \text{diag}(\underline{p}_j^*)\underline{B}'_{0,j}\underline{\beta}_j & , \text{ if } k_j \geq 2. \end{cases}$$

Note that in the above, we defined $\underline{p}_i^*$ as a vector which may contain varying UWC probabilities for each service phase. As we will primarily consider UWC version 2 in this chapter, this is unnecessary for most of our calculations. However, it is left in this generalized form in case one were to consider a model without level-dependent reneging and wished to apply the earlier UWC version 1 results, such as in the example considered in Section 6.6.2.

We begin by considering blocks to track changes in $X_N(t)$. For $n_i = 0, 1, \ldots, C_i$, $i = 1, 2, \ldots, N-1$, we define

$$Q^{[N]}_{n_1,\ldots,n_{N-1}} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C_N-1 \\ C_N \end{array} \begin{array}{c} \begin{array}{cccccc} 0 & 1 & 2 & \cdots & C_N-1 & C_N \end{array} \\ \left[\begin{array}{cccccc} \Delta_{n_1,\ldots,n_{N-1},0} & (UD)^{[N]}_{n_1,\ldots,n_{N-1},0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ (LD)^{[N]}_{n_1,\ldots,n_{N-1},1} & \Delta_{n_1,\ldots,n_{N-1},1} & (UD)^{[N]}_{n_1,\ldots,n_{N-1},1} & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (LD)^{[N]}_{n_1,\ldots,n_{N-1},2} & \Delta_{n_1,\ldots,n_{N-1},2} & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \Delta_{n_1,\ldots,n_{N-1},c_N-1} & (UD)^{[N]}_{n_1,\ldots,n_{N-1},c_N-1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & (LD)^{[N]}_{n_1,\ldots,n_{N-1},c_N} & \Delta_{n_1,\ldots,n_{N-1},c_N} \end{array}\right] \end{array}.$$

The main diagonal blocks of $Q^{[N]}_{n_1,\ldots,n_{N-1}}$ are

$$\Delta_{n_1,\ldots,n_{N-1},n_N} = \begin{bmatrix} \Delta^{[1]}_{n_1,\ldots,n_N} \\ \Delta^{[2]}_{n_1,\ldots,n_N} \\ \vdots \\ \Delta^{[N]}_{n_1,\ldots,n_N} \end{bmatrix},$$

where

$$\Delta^{[1]}_{n_1,\dots,n_N} = \begin{cases} \begin{bmatrix} \zeta_{n_1,\dots,n_N,1} & \underline{S}'_{0,1}\underline{\gamma}_2 & \underline{0}'_{s_1}\underline{0}_{a^{[2,N]}_{n_1,\dots,n_N}-s_2} \end{bmatrix} & , \text{ if } n_1 = 0, \\ \\ \begin{bmatrix} \zeta_{n_1,\dots,n_N,1} & \underline{e}_{|k_1|,1} \otimes \underline{S}'_{0,1}\underline{\beta}_1 & \underline{0}'_{s_1}\underline{0}_{s_2} & \underline{0}'_{s_1}\underline{0}_{a^{[2,N]}_{n_1,\dots,n_N}-s_2} \\ \underline{0}'_{b_1|k_1|}\underline{0}_{s_1} & \zeta_{n_1,\dots,n_N,2} & \zeta^*_{n_1,\dots,n_N,2} & \underline{0}'_{b_1|k_1|}\underline{0}_{a^{[2,N]}_{n_1,\dots,n_N}-s_2} \end{bmatrix} & , \text{ if } n_1 = 1,2,\dots,C_1, \end{cases}$$

while for $j = 2, 3\dots, N-1$,

$$\Delta^{[j]}_{n_1,\dots,n_N} = \begin{cases} \begin{bmatrix} \underline{0}'_{s_j}\underline{0}_{a^{[1,j-1]}_{n_1,\dots,n_N}} & \zeta_{n_1,\dots,n_N,2j-1} & \underline{S}'_{0,j}\underline{\gamma}_{j+1} & \underline{0}'_{s_j}\underline{0}_{a^{[j+1,N]}_{n_1,\dots,n_N}-s_{j+1}} \end{bmatrix} & , \text{ if } n_j = 0, \\ \\ \begin{bmatrix} \underline{0}'_{s_j}\underline{0}_{a^{[1,j-1]}_{n_1,\dots,n_N}} & \zeta_{n_1,\dots,n_N,2j-1} & \underline{e}_{|k_j|,1} \otimes \underline{S}'_{0,j}\underline{\beta}_j & \underline{0}'_{s_j}\underline{0}_{s_{j+1}} & \underline{0}'_{s_j}\underline{0}_{a^{[j+1,N]}_{n_1,\dots,n_N}-s_{j+1}} \\ \underline{0}'_{b_j|k_j|}\underline{0}_{a^{[1,j-1]}_{n_1,\dots,n_N}} & \underline{0}'_{b_j|k_j|}\underline{0}_{s_j} & \zeta_{n_1,\dots,n_N,2j} & \zeta^*_{n_1,\dots,n_N,2j} & \underline{0}'_{b_j|k_j|}\underline{0}_{a^{[j+1,N]}_{n_1,\dots,n_N}-s_{j+1}} \end{bmatrix} & , \text{ if } n_j = 1,2,\dots,C_j, \end{cases}$$

and

$$\Delta^{[N]}_{n_1,\dots,n_N} = \begin{cases} \begin{bmatrix} \underline{S}'_{0,N}\underline{\gamma}_1 & \underline{0}'_{s_N}\underline{0}_{a^{[1,N-1]}_{n_1,\dots,n_N}-s_1} & \zeta_{n_1,\dots,n_N,2N-1} \end{bmatrix} & , \text{ if } n_N = 0, \\ \\ \begin{bmatrix} \underline{0}'_{s_N}\underline{0}_{s_1} & \underline{0}'_{s_N}\underline{0}_{a^{[1,N-1]}_{n_1,\dots,n_N}-s_1} & \zeta_{n_1,\dots,n_N,2N-1} & \underline{e}_{|k_N|,1} \otimes \underline{S}'_{0,N}\beta_N \\ \zeta^*_{n_1,\dots,n_N,2N} & \underline{0}'_{b_N|k_N|}\underline{0}_{a^{[1,N-1]}_{n_1,\dots,n_N}-s_1} & \underline{0}'_{b_N|k_N|}\underline{0}_{s_N} & \zeta_{n_1,\dots,n_N,2N} \end{bmatrix} & , \text{ if } n_N = 1,2,\dots,C_N. \end{cases}$$

The upper diagonal blocks of $Q^{[N]}_{n_1,\dots,n_{N-1}}$ are

$$(UD)^{[N]}_{n_1,\dots,n_{N-1},n_N} = \begin{cases} \begin{bmatrix} \lambda_N I_{a^{[1,N-1]}_{n_1,\dots,n_N}+s_N} & \underline{0}'_{a^{[1,N-1]}_{n_1,\dots,n_N}+s_N}\underline{0}_{b_N|k_N|} \end{bmatrix} & , \text{ if } n_N = 0, \\ \\ \lambda_N I_{a^{[1,N]}_{n_1,\dots,n_N}} & , \text{ if } n_N = 1,2,\dots,C_N-1, \end{cases}$$

and the lower diagonal blocks are

$$(LD)^{[N]}_{n_1,\dots,n_{N-1},n_N} = \begin{bmatrix} \alpha^{[1]}_N(1-\delta_{n_N,C_N}p^*_{N,0})I_{s_1} & \underline{0}'_{s_1}\underline{0}_{a^{[1,N-1]}_{n_1,\dots,n_N}+s_N-s_1} \\ \underline{0}'_{a^{[1,N-1]}_{n_1,\dots,n_N}+s_N-s_1}\underline{0}_{s_1} & \alpha^{[1]}_N(1-\delta_{n_N,C_N}p^*_{N,0})I_{a^{[1,N-1]}_{n_1,\dots,n_N}+s_N-s_1} \\ \underline{e}'_{|k_N|} \otimes (I_{b_N}-\delta_{n_N,C_N}\text{diag}(\underline{p}^*_N))\underline{B}'_{0,N}\underline{\gamma}_1 & \underline{0}'_{b_N|k_N|}\underline{0}_{a^{[1,N-1]}_{n_1,\dots,n_N}+s_N-s_1} \end{bmatrix}$$

for $n_N = 1$ and

$$(LD)^{[N]}_{n_1,\dots,n_{N-1},n_N} =$$

$$\begin{bmatrix} \alpha^{[n_N]}_N(1-\delta_{n_N,C_N}p^*_{N,0})I_{s_1} & \underline{0}'_{s_1}\underline{0}_{a^{[1,N-1]}_{n_1,\dots,n_N}+s_N-s_1} & \underline{0}'_{s_1}\underline{0}_{b_N|k_N|} \\ \underline{0}'_{a^{[1,N-1]}_{n_1,\dots,n_N}+s_N-s_1}\underline{0}_{s_1} & \alpha^{[n_N]}_N(1-\delta_{n_N,C_N}p^*_{N,0})I_{a^{[1,N-1]}_{n_1,\dots,n_N}+s_N-s_1} & \underline{0}'_{a^{[1,N-1]}_{n_1,\dots,n_N}+s_N-s_1}\underline{0}_{b_N|k_N|} \\ U^*_{N,n_N} & \underline{0}'_{b_N|k_N|}\underline{0}_{a^{[1,N-1]}_{n_1,\dots,n_N}+s_N-s_1} & \alpha^{[n_N-1]}_N I_{|k_N|} \otimes (I_{b_N}-\delta_{n_N,C_N}\text{diag}(\underline{p}^*_i)) + U_{N,n_N} \end{bmatrix}$$

for $n_N = 2, 3, \dots, C_N$, where for $i = 1, 2, \dots, N$,

$$U_{i,n_i} = \begin{cases} p^{\text{B}}_i(I_{b_i}-\delta_{n_i,C_i}\text{diag}(\underline{p}^*_i))\underline{B}'_{0,i}\underline{\beta}_i & , \text{ if } k_i = -1, \\ \\ \underline{0}'_{b_i}\underline{0}_{b_i} & , \text{ if } k_i = 1, \\ \\ \begin{bmatrix} \underline{0}'_{k_i-1} & I_{k_i-1} \\ 0 & \underline{0}_{k_i-1} \end{bmatrix} \otimes (I_{b_i}-\delta_{n_i,C_i}\text{diag}(\underline{p}^*_i))\underline{B}'_{0,i}\underline{\beta}_i & , \text{ if } k_i \geq 2, \end{cases}$$

and

$$U^*_{i,n_i} = \begin{cases} (1-p_i^{\mathrm{B}})(I_{b_i} - \delta_{n_i,C_i}\mathrm{diag}(\underline{p}_i^*))\underline{B}'_{0,i}\underline{\gamma}_{i+1} & \text{, if } k_i = -1, \\ \\ \underline{e}'_{k_i,k_i} \otimes (I_{b_i} - \delta_{n_i,C_i}\mathrm{diag}(\underline{p}_i^*))\underline{B}'_{0,i}\underline{\gamma}_{i+1} & \text{, if } k_i \geq 1. \end{cases}$$

Our recursive portion of the generator construction is now defined as follows. For $n_i \in \{0,1,\dots,C_i\}$, $i = 1,2,\dots,j-1$, we define

$$Q^{[j]}_{n_1,\dots,n_{j-1}} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C_j-1 \\ C_j \end{array}\!\!\begin{array}{c} \begin{array}{cccccc} 0 & 1 & 2 & \cdots & C_j-1 & C_j \end{array} \\ \left[\begin{array}{cccccc} Q^{[j+1]}_{n_1,\dots,n_{j-1},0} & (UD)^{[j]}_{n_1,\dots,n_{j-1},0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ (LD)^{[j]}_{n_1,\dots,n_{j-1},1} & Q^{[j+1]}_{n_1,\dots,n_{j-1},1} & (UD)^{[j]}_{n_1,\dots,n_{j-1},1} & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (LD)^{[j]}_{n_1,\dots,n_{j-1},2} & Q^{[j+1]}_{n_1,\dots,n_{j-1},2} & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q^{[j+1]}_{n_1,\dots,n_{j-1},c_j-1} & (UD)^{[j]}_{n_1,\dots,n_{j-1},c_j-1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & (LD)^{[j]}_{n_1,\dots,n_{j-1},c_j} & Q^{[j+1]}_{n_1,\dots,n_{j-1},c_j} \end{array}\right] \end{array}.$$

The upper diagonal blocks $(UD)^{[j]}_{n_1,n_2,\dots,n_{j-1},n_j}$ take the form of Equation (5.46) (with $k = 0$), where

$$(UD)^{[j]}_{n_1,\dots,n_{N-1},n_N} = \begin{bmatrix} \lambda_j I_{a^{[1,j-1]}_{n_1,\dots,n_N}+s_j} & \underline{0}'_{a^{[1,j-1]}_{n_1,\dots,n_N}+s_j}\underline{0}_{b_j|k_j|} & \underline{0}'_{a^{[1,j-1]}_{n_1,\dots,n_N}+s_j}\underline{0}_{a^{[j+1,N]}_{n_1,\dots,n_N}} \\ \underline{0}'_{a^{[j+1,N]}_{n_1,\dots,n_N}}\underline{0}_{a^{[1,j-1]}_{n_1,\dots,n_N}+s_j} & \underline{0}'_{a^{[j+1,N]}_{n_1,\dots,n_N}}\underline{0}_{b_j|k_j|} & \lambda_j I_{a^{[j+1,N]}_{n_1,\dots,n_N}} \end{bmatrix}$$

for $n_j = 0$ and

$$(UD)^{[j]}_{n_1,\dots,n_{N-1},n_N} = \lambda_j I_{a^{[1,N]}_{n_1,\dots,n_N}}$$

for $n_j = 1,2,\dots,C_j-1$. Similarly, the lower diagonal blocks $(LD)^{[j]}_{n_1,n_2,\dots,n_{j-1},n_j}$ satisfy Equation (5.47) (with $k = 0$), where

$$(LD)^{[j]}_{n_1,\dots,n_{N-1},n_N} = \begin{bmatrix} \alpha_j^{[1]}(1-\delta_{n_j,C_j}p^*_{j,0})I_{a^{[1,j-1]}_{n_1,\dots,n_N}+s_j} & \underline{0}'_{a^{[1,j-1]}_{n_1,\dots,n_N}+s_j}\underline{0}_{s_{j+1}} & \underline{0}'_{a^{[1,j-1]}_{n_1,\dots,n_N}+s_j}\underline{0}_{a^{[j+1,N]}_{n_1,\dots,n_N}-s_{j+1}} \\ \underline{0}'_{b_j|k_j|}\underline{0}_{a^{[1,j-1]}_{n_1,\dots,n_N}+s_j} & \underline{e}'_{|k_j|} \otimes (I_{b_j}-\delta_{n_j,C_j}\mathrm{diag}(\underline{p}_j^*))\underline{B}'_{0,j}\underline{\gamma}_{j+1} & \underline{0}'_{b_j|k_j|}\underline{0}_{a^{[j+1,N]}_{n_1,\dots,n_N}-s_{j+1}} \\ \underline{0}'_{s_{j+1}}\underline{0}_{a^{[1,j-1]}_{n_1,\dots,n_N}+s_j} & \alpha_j^{[1]}(1-\delta_{n_j,C_j}p^*_{j,0})I_{s_{j+1}} & \underline{0}'_{s_{j+1}}\underline{0}_{a^{[j+1,N]}_{n_1,\dots,n_N}-s_{j+1}} \\ \underline{0}'_{a^{[j+1,N]}_{n_1,\dots,n_N}-s_{j+1}}\underline{0}_{a^{[1,j-1]}_{n_1,\dots,n_N}+s_j} & \underline{0}'_{a^{[j+1,N]}_{n_1,\dots,n_N}-s_{j+1}}\underline{0}_{s_{j+1}} & \alpha_j^{[1]}(1-\delta_{n_j,C_j}p^*_{j,0})I_{a^{[j+1,N]}_{n_1,\dots,n_N}-s_{j+1}} \end{bmatrix}$$

for $n_j = 1$ and

$$(LD)^{[j]}_{n_1,\dots,n_{N-1},n_N} =$$

$$\begin{bmatrix} \alpha_j^{[n_j]}(1-\delta_{n_j,C_j}p^*_{j,0})I_{a^{[1,j-1]}_{n_1,\dots,n_N}+s_j} & \underline{0}'_{a^{[1,j-1]}_{n_1,\dots,n_N}+s_j}\underline{0}_{b_j|k_j|} & \underline{0}'_{a^{[1,j-1]}_{n_1,\dots,n_N}+s_j}\underline{0}_{s_{j+1}} & \underline{0}'_{a^{[1,j-1]}_{n_1,\dots,n_N}+s_j}\underline{0}_{a^{[j+1,N]}_{n_1,\dots,n_N}-s_{j+1}} \\ \underline{0}'_{b_j|k_j|}\underline{0}_{a^{[1,j-1]}_{n_1,\dots,n_N}+s_j} & \alpha_j^{[n_j-1]}I_{|k_j|} \otimes (I_{b_j}-\delta_{n_j,C_j}\mathrm{diag}(\underline{p}_j^*))+U_{j,n_j} & U^*_{j,n_j} & \underline{0}'_{b_j|k_j|}\underline{0}_{a^{[j+1,N]}_{n_1,\dots,n_N}-s_{j+1}} \\ \underline{0}'_{s_{j+1}}\underline{0}_{a^{[1,j-1]}_{n_1,\dots,n_N}+s_j} & \underline{0}'_{s_{j+1}}\underline{0}_{b_j|k_j|} & \alpha_j^{[n_j]}(1-\delta_{n_j,C_j}p^*_{j,0})I_{s_{j+1}} & \underline{0}'_{s_{j+1}}\underline{0}_{a^{[j+1,N]}_{n_1,\dots,n_N}-s_{j+1}} \\ \underline{0}'_{a^{[j+1,N]}_{n_1,\dots,n_N}-s_{j+1}}\underline{0}_{a^{[1,j-1]}_{n_1,\dots,n_N}+s_j} & \underline{0}'_{a^{[j+1,N]}_{n_1,\dots,n_N}-s_{j+1}}\underline{0}_{b_j|k_j|} & \underline{0}'_{a^{[j+1,N]}_{n_1,\dots,n_N}-s_{j+1}}\underline{0}_{s_{j+1}} & \alpha_j^{[n_j]}(1-\delta_{n_j,C_j}p^*_{j,0})I_{a^{[j+1,N]}_{n_1,\dots,n_N}-s_{j+1}} \end{bmatrix}$$

for $n_j = 2,3,\dots,C_j$.

Lastly, the complete infinitesimal generator matrix $Q$ is the QBD describing changes in the level of the process, $X_1(t)$, taking the form

$$Q = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ C_1-1 \\ C_1 \end{array} \begin{array}{c} \begin{array}{cccccc} 0 & 1 & 2 & \cdots & C_1-1 & C_1 \end{array} \\ \left[\begin{array}{cccccc} Q_0^{[2]} & (UD)_0^{[1]} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ (LD)_1^{[1]} & Q_1^{[2]} & (UD)_1^{[1]} & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (LD)_2^{[1]} & Q_2^{[2]} & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & Q_{C_1-1}^{[2]} & (UD)_{C_1-1}^{[1]} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & (LD)_{C_1}^{[1]} & Q_{C_1}^{[2]} \end{array}\right] \end{array},$$

where we let

$$(UD)^{[1]}_{n_1,\ldots,n_{N-1},n_N} = \begin{bmatrix} \lambda_1 I_{s_1} & \underline{0}'_{s_1}\underline{0}_{b_1|k_1|} & \underline{0}'_{s_1}\underline{0}_{a^{[2,N]}_{n_1,\ldots,n_N}} \\ \underline{0}'_{a^{[2,N]}_{n_1,\ldots,n_N}}\underline{0}_{s_1} & \underline{0}'_{a^{[2,N]}_{n_1,\ldots,n_N}}\underline{0}_{b_1|k_1|} & \lambda_1 I_{a^{[2,N]}_{n_1,\ldots,n_N}} \end{bmatrix}$$

for $n_1 = 0$ and

$$(UD)^{[1]}_{n_1,\ldots,n_{N-1},n_N} = \lambda_1 I_{a^{[1,N]}_{n_1,\ldots,n_N}}$$

for $n_1 = 1, 2, \ldots, C_1 - 1$, and

$$(LD)^{[1]}_{n_1,\ldots,n_{N-1},n_N} = \begin{bmatrix} \alpha_1^{[1]}(1-\delta_{n_1,C_1}p^*_{1,0})I_{s_1} & \underline{0}'_{s_1}\underline{0}_{s_2} & \underline{0}'_{s_1}\underline{0}_{a^{[2,N]}_{n_1,\ldots,n_N}-s_2} \\ \underline{0}'_{b_1|k_1|}\underline{0}_{s_1} & \underline{e}'_{|k_1|} \otimes (I_{b_1}-\delta_{n_1,C_1}\mathrm{diag}(\underline{p}^*_1))\underline{B}'_{0,1}\underline{\gamma}_2 & \underline{0}'_{b_1|k_1|}\underline{0}_{a^{[2,N]}_{n_1,\ldots,n_N}-s_2} \\ \underline{0}'_{s_2}\underline{0}_{s_1} & \alpha_1^{[1]}(1-\delta_{n_1,C_1}p^*_{1,0})I_{s_2} & \underline{0}'_{s_2}\underline{0}_{a^{[2,N]}_{n_1,\ldots,n_N}-s_2} \\ \underline{0}'_{a^{[2,N]}_{n_1,\ldots,n_N}-s_2}\underline{0}_{s_1} & \underline{0}'_{a^{[2,N]}_{n_1,\ldots,n_N}-s_2}\underline{0}_{s_2} & \alpha_1^{[1]}(1-\delta_{n_1,C_1}p^*_{1,0})I_{a^{[2,N]}_{n_1,\ldots,n_N}-s_2} \end{bmatrix}$$

for $n_1 = 1$ and

$$(LD)^{[1]}_{n_1,\ldots,n_{N-1},n_N} =$$

$$\begin{bmatrix} \alpha_1^{[n_1]}(1-\delta_{n_1,C_1}p^*_{1,0})I_{s_1} & \underline{0}'_{s_1}\underline{0}_{b_1|k_1|} & \underline{0}'_{s_1}\underline{0}_{s_2} & \underline{0}'_{s_1}\underline{0}_{a^{[2,N]}_{n_1,\ldots,n_N}-s_2} \\ \underline{0}'_{b_1|k_1|}\underline{0}_{s_1} & \alpha_1^{[n_1-1]}I_{|k_1|}(I_{b_1}-\delta_{n_1,C_1}\mathrm{diag}(\underline{p}^*_1))+U_{1,n_1} & U^*_{1,n_1} & \underline{0}'_{b_1|k_1|}\underline{0}_{a^{[2,N]}_{n_1,\ldots,n_N}-s_2} \\ \underline{0}'_{s_2}\underline{0}_{s_1} & \underline{0}'_{s_2}\underline{0}_{b_1|k_1|} & \alpha_1^{[n_1]}(1-\delta_{n_1,C_1}p^*_{1,0})I_{s_2} & \underline{0}'_{s_2}\underline{0}_{a^{[2,N]}_{n_1,\ldots,n_N}-s_2} \\ \underline{0}'_{a^{[2,N]}_{n_1,\ldots,n_N}-s_2}\underline{0}_{s_1} & \underline{0}'_{a^{[2,N]}_{n_1,\ldots,n_N}-s_2}\underline{0}_{b_1|k_1|} & \underline{0}'_{a^{[2,N]}_{n_1,\ldots,n_N}-s_2}\underline{0}_{s_2} & \alpha_1^{[n_1]}(1-\delta_{n_1,C_1}p^*_{1,0})I_{a^{[2,N]}_{n_1,\ldots,n_N}-s_2} \end{bmatrix}$$

for $n_1 = 2, 3, \ldots, C_1$.

## 6.4 Calculation of UWC Probabilities

### 6.4.1 When the Server is Away

When the server is away from $Q_i$, either conducting a switch or serving a different queue, $X_i(t)$ in the IB model will develop in time as a normal birth-and-death process having constant birth rate $\lambda_i$ and a death rate of $\alpha_i^{[j]}$ when $X_i(t) = j$, $j \in \mathbb{Z}^+$. Thus, the balance equations of this birth-and-death model are simply

$$\lambda_i \pi_j = \alpha_i^{[j+1]}\pi_{j+1}, \; j \in \mathbb{N}.$$

The duration of a level-$C$ busy period in this CTMC is identical in distribution to that of a standard busy period in a birth-and-death process having the indices of its birth and death rates shifted by $C-1$. That is, the busy period of a CTMC having balance equations

$$\lambda_i \pi_j = \alpha_i^{[C+j]} \pi_{j+1},\ j \in \mathbb{N}.$$

If we make the reasonable assumption that that there exists an $n \in \mathbb{Z}^+$ such that $\alpha_i^{[j]} > \lambda_i$, $\forall\ j \geq n$, then this CTMC is ergodic and has steady-state distribution (e.g., Ross [82], p. 376)

$$\pi_j = \frac{\lambda_i^j \left(\prod_{k=0}^{j-1} \alpha_i^{[C+k]}\right)^{-1}}{1 + \sum_{n=1}^{\infty} \lambda_i^n \left(\prod_{k=0}^{n-1} \alpha_i^{[C+k]}\right)^{-1}},\ j \in \mathbb{N},$$

where we let $\prod_{k=0}^{0-1} \alpha^{[C+k]} = 1$. It follows by Equation (1.10) that for the shifted process, we have

$$\begin{aligned}
\mathrm{E}[BP] &= \frac{1-\pi_0}{\lambda_i \pi_0} \\
&= \left(1 - \frac{1}{1 + \sum_{n=1}^{\infty} \lambda_i^n \left(\prod_{k=0}^{n-1} \alpha_i^{[C+k]}\right)^{-1}}\right) \left(\frac{\lambda_i}{1 + \sum_{n=1}^{\infty} \lambda_i^n \left(\prod_{k=0}^{n-1} \alpha_i^{[C+k]}\right)^{-1}}\right)^{-1} \\
&= \sum_{n=1}^{\infty} \lambda_i^{n-1} \left(\prod_{k=0}^{n-1} \alpha_i^{[C+k]}\right)^{-1}.
\end{aligned}$$

The distribution of a 'service time' of the shifted CTMC is $\mathrm{Exp}(\alpha_i^{[C]})$ (where we may attribute any excess in death rate at higher queue lengths to reneging), and thus has an expected value of $(\alpha_i^{[C]})^{-1}$. Therefore, for class $i$ we set

$$\frac{1}{1-p_{i,0}^*} = \frac{\sum_{n=1}^{\infty} \lambda_i^{n-1} \left(\prod_{k=0}^{n-1} \alpha_i^{[C+k]}\right)^{-1}}{(\alpha_i^{[C]})^{-1}} = \sum_{n=1}^{\infty} \frac{\lambda_i^{n-1} \alpha_i^{[C]}}{\prod_{k=0}^{n-1} \alpha_i^{[C+k]}},$$

implying that

$$p_{i,0}^* = 1 - \left(\sum_{n=1}^{\infty} \frac{\lambda_i^{n-1} \alpha_i^{[C]}}{\prod_{k=0}^{n-1} \alpha_i^{[C+k]}}\right)^{-1},\ i = 1, 2, \ldots, N.$$

### 6.4.2 When the Server is Visiting

To handle the UWC approximation in a queue during a server's visit, we simply generalize the analysis of Sections 5.6.2 and 5.7.1 to handle level-dependent reneging. No adjustment needs to be made for the change in service discipline, as the distribution of the *required* number of observed departures to decrease the observed queue length does not change if the server stops serving prior to this time. As we are not adding any states to indicate how long the queue has been at observable capacity (e.g., reaching it during a visit or while the server is away), any excess waiting customers are effectively 'lost' when the UWC probability at a queue is changed, and so the discipline should not impact the probability of an unobserved waiting customer being present after an observed departure.

For a given truncation $D$ on the number of modelled unobserved waiting customers beyond queue position $C$, as well as probability row vector $\underline{\beta}^*_C$ for the initial service phase at the start of the level-$C$ busy period, the (approximate) PMF of the number of unobserved waiting customers that immediately replace observed departures in a level-$C$ busy period again takes the form

$$P(N^*_C = n) = \left[\begin{array}{cc} \underline{\beta}^*_C & \underline{0} \end{array}\right] \left[(-Q^{-1}_{TT})Q^*_{TA}\right]^n (-Q^{-1}_{TT})\underline{Q}'_{-1},\ n \in \mathbb{N},$$

where we now let for a given $i = 1, 2, \ldots, N$ (which we suppress), $\Delta_j = B - (\lambda + \alpha^{[C-1+j]})I_b$, $j = 0, 1, \ldots, D$, $\Delta_A = \underline{B}'_0\underline{\beta} + \alpha^{[C-1]}I_b$, $\alpha^{[C-1,j]} = \alpha^{[C-1+j]} - \alpha^{[C-1]}$, $j = 1, 2, \ldots, D$,

$$Q_{TT} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ D-1 \\ D \end{array} \begin{array}{c} \begin{array}{cccccc} 0 & 1 & 2 & \cdots & D-1 & D \end{array} \\ \left[\begin{array}{cccccc} \Delta_0 & \lambda I_b & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \alpha^{[C-1,1]}I_b & \Delta_1 & \lambda I_b & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \alpha^{[C-1,2]}I_b & \Delta_2 & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \Delta_{D-1} & \lambda I_b \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \alpha^{[C-1,D]}I_b & \Delta_D + \lambda I_b \end{array}\right] \end{array},$$

$$Q^*_{TA} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ D-1 \\ D \end{array} \begin{array}{c} \begin{array}{cccccc} 0' & 1' & \cdots & (D-2)' & (D-1)' & D' \end{array} \\ \left[\begin{array}{cccccc} \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \underline{0}'_b\underline{0}_b \\ \Delta_A & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \Delta_A & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \Delta_A & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \Delta_A & \mathbf{0} \end{array}\right] \end{array},$$

and

$$\underline{Q}'_{-1} = \left[\begin{array}{c} \Delta_A\underline{e}' \\ \underline{0}' \end{array}\right].$$

From the PMF of $N^*_C$, we calculate $\mathrm{E}[N^*_C]$ and set

$$\mathrm{E}[N^*_C] = \frac{p^*_C}{1 - p^*_C},$$

or

$$p^*_{i,j} = \frac{\mathrm{E}[N^*_{C,i}]}{1 + \mathrm{E}[N^*_{C,i}]},\ j = 1, 2, \ldots, b_i,$$

where we let $\mathrm{E}[N^*_{C,i}]$ represent the expected number of unobserved waiting customers that immediately replace observed departures in a class-$i$ level-$C$ busy period using $\underline{\beta}^*_C = \hat{\underline{\beta}}^*_i$, which is calculated using FB model probabilities in place of IB model probabilities in Equation (5.44), extended logically to handle the summation over state index $k$.

## 6.5 Nominal Waiting Time, Time Spent Waiting, and the Probability of Reneging

We are interested in the distribution of the waiting time of a class-1 customer as well as the probability of them reaching service. Equivalent results for other classes may be obtained by shifting class indices, such that a class of interest is treated as class 1, while maintaining the appropriate relative cyclic polling order. In Remark 4.2, it was stated that if we do not treat a truncated system as having a finite buffer, then it is more accurate to allow a class-1 customer to stay in the system even if they observe $X_1(t) = C_1$ at their arrival instant. In this case, we assume that they are waiting in position $C_1 + 1$ as we are not modelling the exact number of unobserved waiting customers. While not without some degree of inaccuracy when approximating the true IB model waiting time distribution, it is more appropriate than using the waiting time distribution of a customer conditional on them arriving into an observable queue position.

If we suppose that the target class-1 customer is patient, then their nominal waiting time until reaching service can be modelled by the absorbing CTMC

$$\begin{bmatrix} \mathcal{R} & \underline{\mathcal{R}}'_0 \\ \underline{0}_\ell & 0 \end{bmatrix},$$

where $\ell$ is the total number of states in Equation (6.1) since we are not removing level $C_1$, and $\underline{\mathcal{R}}'_0 = -\mathcal{R}\underline{e}'$. Here, we are letting

$$\mathcal{R} = \begin{array}{c} \\ C_1 \\ C_1-1 \\ C_1-2 \\ \vdots \\ 2 \\ 1 \\ 0 \end{array} \begin{array}{c} \begin{array}{ccccccc} C_1 & C_1-1 & C_1-2 & \cdots & 2 & 1 & 0 \end{array} \\ \left[\begin{array}{ccccccc} \widetilde{Q}^{[2]}_{C_1} & (LD)^{[1]}_{C_1} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \widetilde{Q}^{[2]}_{C_1-1} & (LD)^{[1]}_{C_1-1} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \widetilde{Q}^{[2]}_{C_1-2} & \ddots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \widetilde{Q}^{[2]}_{2} & (LD)^{[1]}_{2} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \widetilde{Q}^{[2]}_{1} & (\widetilde{LD})^{[1]}_{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \widetilde{Q}^{[2]}_{0} \end{array}\right] \end{array},$$

where the level now represents the number of class-1 customers ahead of our target customer (in service or waiting).

As further class-1 arrivals will not impact the waiting time of a class-1 customer already in the system, we turn off the flow of class-1 arrivals and let $\widetilde{Q}^{[2]}_m = Q^{[2]}_m + \lambda_1(1 - \delta_{m,C_1})I$, $m = 1, 2, \dots, C_1$. Next, to account for the possibility of the target customer entering service after the customer immediately preceding them completes service, we have a modified block $(\widetilde{LD})^{[1]}_1$ which is constructed as per Equation (5.47) but using

$$(\widetilde{LD})^{[1]}_{n_1=1,\dots,n_{N-1},n_N} = \begin{bmatrix} \alpha^{[1]}_1(1-\delta_{n_1,C_1}p^*_{1,0})I_{s_1} & \underline{0}'_{s_1}\underline{0}_{s_2} & \underline{0}'_{s_1}\underline{0}_{a^{[2,N]}_{n_1,\dots,n_N}-s_2} \\ \underline{0}'_{b_1|k_1|}\underline{0}_{s_1} & U^*_{1,n_1} & \underline{0}'_{b_1|k_1|}\underline{0}_{a^{[2,N]}_{n_1,\dots,n_N}-s_2} \\ \underline{0}'_{s_2}\underline{0}_{s_1} & \alpha^{[1]}_1(1-\delta_{n_1,C_1}p^*_{1,0})I_{s_2} & \underline{0}'_{s_2}\underline{0}_{a^{[2,N]}_{n_1,\dots,n_N}-s_2} \\ \underline{0}'_{a^{[2,N]}_{n_1,\dots,n_N}-s_2}\underline{0}_{s_1} & \underline{0}'_{a^{[2,N]}_{n_1,\dots,n_N}-s_2}\underline{0}_{s_2} & \alpha^{[1]}_1(1-\delta_{n_1,C_1}p^*_{1,0})I_{a^{[2,N]}_{n_1,\dots,n_N}-s_2} \end{bmatrix}.$$

Finally, we must also make an adjustment to account for the fact that in level 0, after observing a class-1 switch-in time completion, the target customer enters service rather than the server immediately beginning a class-2 switch-in. To this end, we use the modified block $\widetilde{Q}_0^{[2]}$ which is calculated as $Q_0^{[2]} + \lambda_1 I$, but we replace $\Delta^{[1]}_{n_1=0,\ldots,n_N}$ by

$$\widetilde{\Delta}^{[1]}_{n_1=0,\ldots,n_N} = \left[ \begin{array}{ccc} \zeta_{n_1,\ldots,n_N,1} & \underline{0}'_{s_1}\underline{0}_{s_2} & \underline{0}'_{s_1}\underline{0}_{a^{[2,N]}_{n_1,\ldots,n_N}-s_2} \end{array} \right].$$

It then follows that the nominal waiting time $W_1^*$ is phase-type with representation $\mathrm{PH}_\ell(\underline{\Phi}, \mathcal{R})$, using initial probability row vector

$$\underline{\Phi} = (\underline{\pi}_{C_1}, \underline{\pi}_{C_1-1}, \ldots, \underline{\pi}_1, \underline{\pi}_0).$$

We now incorporate the target customer's reneging to obtain the time spent waiting in the system from what we have already constructed for the nominal waiting time. Treating 'reaching service' and 'reneging' as separate absorbing states, their time spent waiting in the system is equal in distribution to the time until absorption for a CTMC with infinitesimal generator matrix

$$\left[ \begin{array}{ccc} \mathcal{R} - \mathcal{A}^{[1]} & \underline{\mathcal{R}}'_0 & \underline{\alpha}'_1 \\ \underline{0}_\ell & 0 & 0 \\ \underline{0}_\ell & 0 & 0 \end{array} \right],$$

and initial probability vector $(\underline{\Phi}, 0, 0)$. Here, we are letting $\underline{\alpha}'_1 = \mathcal{A}^{[1]}\underline{e}'$ be the ordered column vector of the target class-1 customer's reneging rates and $\mathcal{A}^{[1]}$ is a square matrix which places these ordered rates on the main diagonal. $\mathcal{A}^{[1]}$ is constructed as

$$\mathcal{A}^{[1]} = \left[ \begin{array}{cccc} \mathcal{A}^{[1]}_{C_1} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathcal{A}^{[1]}_{C_1-1} & \ddots & \mathbf{0} \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathcal{A}^{[1]}_0 \end{array} \right],$$

whose blocks we obtain recursively from the following, such that $\mathcal{A}^{[1]}_{n_1,\ldots,n_{1+0}} = \mathcal{A}^{[1]}_{n_1}$:

$$\mathcal{A}^{[1]}_{n_1,\ldots,n_{1+k}} = \left[ \begin{array}{cccc} \mathcal{A}^{[1]}_{n_1,\ldots,n_{1+k},0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathcal{A}^{[1]}_{n_1,\ldots,n_{1+k},1} & \ddots & \mathbf{0} \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathcal{A}^{[1]}_{n_1,\ldots,n_{1+k},C_{1+k+1}} \end{array} \right], \ k = 0, 1, \ldots, N-2.$$

The block components of the above are

$$\mathcal{A}^{[1]}_{n_1,\ldots,n_N} = \left[ \begin{array}{ccc} \alpha_{1,n_1+1} I_{s_1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \alpha_{1,n_1} I_{|k_1|b_1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \alpha_{1,n_1+1} I_{a^{[2,N]}_{n_1,\ldots,n_N}} \end{array} \right]$$

for $n_1 = 1, 2, \ldots, C_1 - 1$, and

$$\mathcal{A}^{[1]}_{n_1,\ldots,n_N} = \alpha_{1,1} I_{a^{[1,N]}_{n_1,\ldots,n_N}}$$

for $n_1 = 0$. It then follows that the customer's time spent waiting in the system $W_1^{\#}$ follows a $\text{PH}_\ell(\Phi, \mathcal{R} - \mathcal{A}^{[1]})$ distribution and has expected value

$$\text{E}[W_i^{\#}] = \underline{\Phi}(\mathcal{A}^{[1]} - \mathcal{R})^{-1}\underline{e}'.$$

Lastly, as in Equation (4.9), it follows that

$$P(\text{Reach Service}) = \underline{\Phi}(\mathcal{A}^{[1]} - \mathcal{R})^{-1}\underline{\mathcal{R}}_0'$$

and

$$P(\text{Renege}) = \underline{\Phi}(\mathcal{A}^{[1]} - \mathcal{R})^{-1}\underline{\alpha}_1'.$$

## 6.6 Numerical Examples

### 6.6.1 Waiting Time Densities

We examine densities of both class-1 nominal waiting time and time spent waiting for FB and UWC models at increasing values of $C$. A value of $D = 50$ was used to calculate the UWC probabilities, as outlined in Section 6.4.2. We consider a 2-queue system with arrival rates $\lambda_1 = \lambda_2 = 0.75$, Exp(10) switch-in times for both classes, and exponentially distributed service times with means of 1 and 0.5 for classes 1 and 2, respectively. Additionally, we suppose that there is a Bernoulli service discipline at each queue, with $p_1^{\text{B}} = 0.25$ and $p_2^{\text{B}} = 1$, such that class 2 is served exhaustively.

For $C_1 = C_2 = C$, $C = 2, 3, \ldots, 25$, we plot the densities of both the nominal waiting time as well as time spent waiting distributions outlined in Section 6.5. In Figure 6.1, we assume simple level-independent reneging rates

$$\alpha_{i,n} = 0.025, \;\; i = 1, 2,$$

while in Figure 6.2, we allow the reneging rates to scale linearly with queue position,

$$\alpha_{i,n} = 0.01 + 0.005n, \;\; i = 1, 2.$$

Along with the densities for both FB and UWC models, we plot the waiting times' expected values and variances for given $C$ values. Note that the horizontal grey lines in Figures 6.1 and 6.2, (e) and (f), are approximations of the limiting IB model expected values and variances approximated using the corresponding FB models with $C = 40$.

Recall that the time spent waiting in queue is in effect, the minimum of the customer's nominal waiting time and their impatience time. As such, it should be no greater than the nominal waiting time. Comparing the nominal waiting time and time spent waiting densities at $C = 25$ in either figure, we observe this through a significant increase in density at smaller values of $t$. In particular, this is more noticeable in Figure 6.2 where reneging rates are larger for customers having 3 or more customers waiting ahead of them in their queue. Interestingly, this also results in the level-independent reneging model having smaller average time spent waiting than the level-dependent reneging model for $C \leq 5$ in the UWC model and $C \leq 6$ in the FB model.

In both figures, it is clear that the FB and UWC models converge to the same densities as we increase $C$. At small $C$ the densities differ greatly, with the UWC curves being quicker to lower the density at the smallest values of $t$. Also, for example, we observe at $C = 2$ that the densities flatten out near 0 at slightly higher values of $t$, indicating larger waiting times. Through parts (e) and (f) of either figure, we demonstrate that the expected values and variances of the UWC models are strictly larger than those of the FB models at a given $C$, while monotonically converging to the same IB model limit. This stands as evidence that the UWC models provide better approximations of the true densities than the FB models at a given truncation level $C$.

### 6.6.2 Revisiting a Cost Optimization Problem

Within this example, we investigate the impact of UWC on the selection of optimal $(k_1, k_2)$ pairs as considered in Section 4.5.1. As we previously found optimality along the boundary $k_1 + k_2 = 12$, we will now only consider this subset of possible pairs. In particular, we consider the parameters of Case 1 such that in our 2-queue system we have arrival rates $\lambda_1 = \lambda_2 = 0.75$, Exp(10) switch-in time distributions, and mean service times of $\mu_1 = 0.9$ and $\mu_2 = 0.1$ for classes 1 and 2, respectively. As we are interested in the impact of UWC, we select the $\mathrm{H}_2$ service time distributions, having variances of $1000\mu_i^2$:

$$Ser_i \sim \mathrm{PH}_2\left(\underline{\beta}_i = (0.001, 0.999), B_i = \begin{bmatrix} -\left(\frac{1}{\mu_i}\right)\left(\frac{\sqrt{2}}{\sqrt{2}+999}\right) & 0 \\ 0 & -\left(\frac{1}{\mu_i}\right)\left(\frac{\sqrt{2}}{\sqrt{2}-1}\right) \end{bmatrix}\right),\ i = 1, 2.$$

As this example considered level-independent reneging rates, we may apply UWC version 1 as covered in Chapter 5, in addition to UWC version 2. A value of $D = 50$ was used in the UWC version 2 calculations.

In Figure 6.3, we plot optimal $k_1$ (such that $k_2 = 12 - k_1$, $k_1 = 1, 2, \ldots, 11$) for FB and both UWC models, letting $\alpha_1 = \alpha_2 = \alpha \in \{0.025, 0.05, 0.25\}$ and $C_1 = C_2 = C$, $C = 2, 3, \ldots, 20$. To aid in visualization, as we are plotting discrete values on our vertical axis, we make use of a small vertical displacement on the FB and UWC version 2 values so that the data points do not overlap. Here, optimal $k_1$ are defined as those satisfying our restrictions that minimize the cost function

$$\mathrm{Cost} = \mathrm{Cost}_1 + \mathrm{Cost}_2,$$

where

$$\mathrm{Cost}_i = c_i\lambda_i\mathrm{E}[W_i^{\#}] + r_i\lambda_i\mathrm{Pr}(\text{Class-}i\text{ customer reneges}),\ i = 1, 2,$$

and $c_i$ and $r_i$, $i = 1, 2$, are assumed to be non-negative constants, which for the purposes of this example are set equal to $c_1 = 2$, $c_2 = 1$, $r_1 = 1$, and $r_2 = 0.5$.

Figure 6.3 provides us with an idea on how fast (in terms of $C$) each model arrives upon the correct optimal $k_1$ values, as well as the volatility of these values. In Figure 6.3 (a), we see that all three models arrive on $k_1 = 3$ at $C = 8$, however UWC version 2 diverts to $k_1 = 4$ for a period before eventually returning. At smaller $C$, FB overshoots the optimal $k_1$ and arrives on $k_1 = 5$. In Figure 6.3 (b), the models all hit $k_1 = 3$ at $C = 8$ again. While UWC version 2 does not divert from this value, we again observe the FB model reach $k_1 = 5$ at small $C$ while neither UWC model exceeds $k_1 = 4$. Lastly, in Figure 6.3 (c), UWC version 1 begins in the correct optimal $k_1$ and never leaves, while version 2 is the next to arrive from below, followed by FB from above. Overall, within these three cases, we observe UWC version 1 to be the most reliable, reaching the final optimal $k_1$ values no slower than the other models, while having minimal volatility.

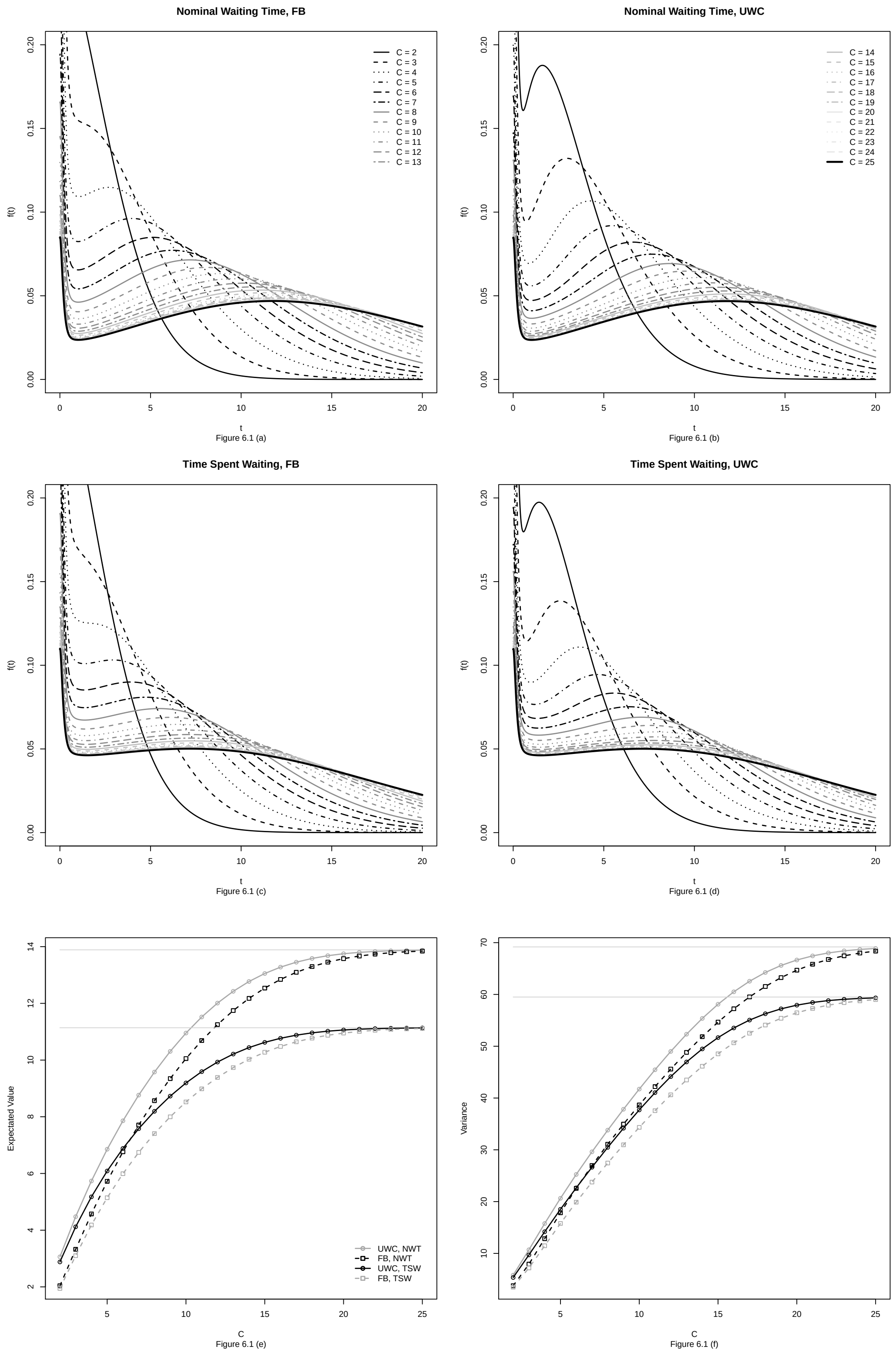

Figure 6.1 (a)

Figure 6.1 (b)

Figure 6.1 (c)

Figure 6.1 (d)

Figure 6.1 (e)

Figure 6.1 (f)

Figure 6.1: Plots of FB and UWC waiting time densities, expected values, and variances, for $C = 2, 3, \ldots, 25$, and $\alpha_{i,n} = 0.025$, $i = 1, 2$.

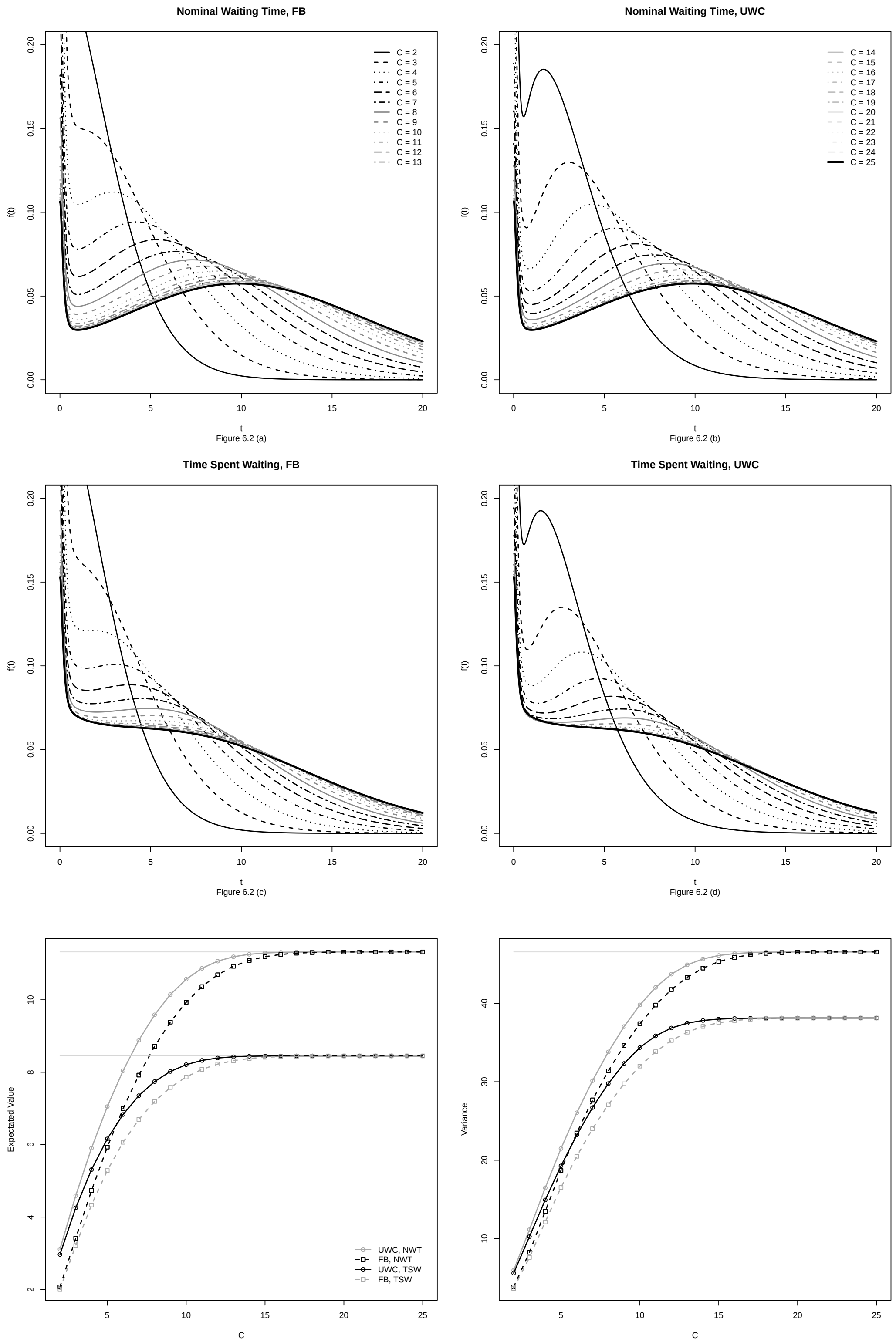

Figure 6.2 (a)

Figure 6.2 (b)

Figure 6.2 (c)

Figure 6.2 (d)

Figure 6.2 (e)

Figure 6.2 (f)

Figure 6.2: Plots of FB and UWC waiting time densities, expected values, and variances, for $C = 2, 3, \ldots, 25$, and $\alpha_{i,n} = 0.01 + 0.005n$, $i = 1, 2$.

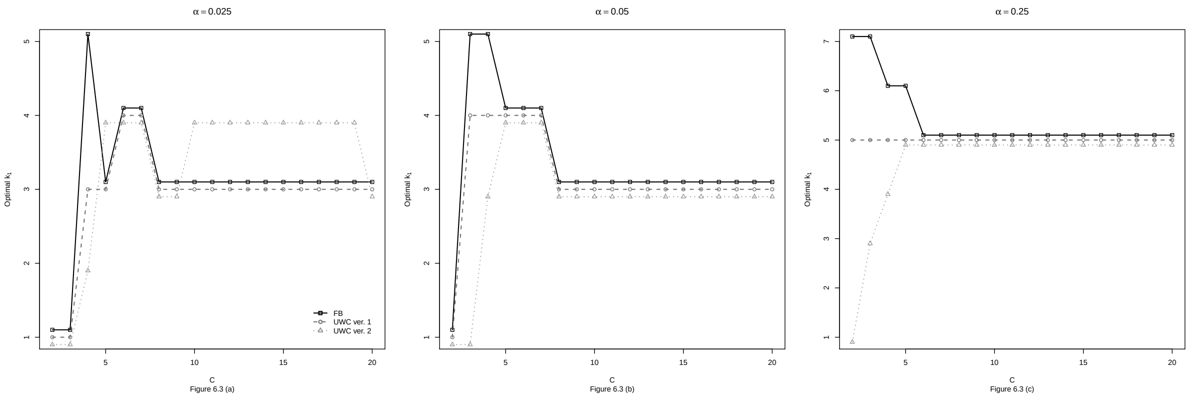

Figure 6.3: Plots of optimal $k_1$ at $C = 2, 3, \ldots, 20$ for FB, UWC version 1, and UWC version 2 models.

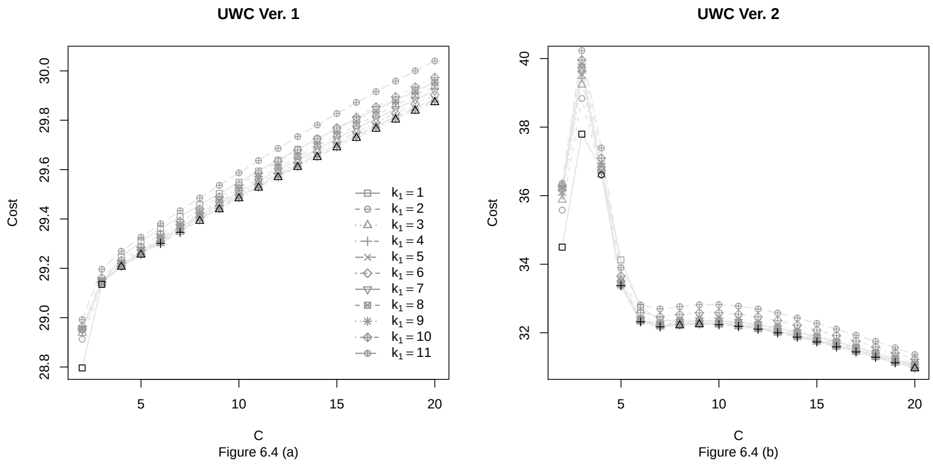

Figure 6.4: Plots of the cost function at $C = 2, 3, \ldots, 20$ and $k_1 = 1, 2, \ldots, 11$, for UWC version 1 and 2 models at $\alpha = 0.025$.

Previously, we have observed UWC version 2 outperform UWC version 1. To see why this is not the case in this particular example, we plot the cost functions for $\alpha = 0.025$ at each $k_1 = 1, 2, \ldots, 11$ in Figure 6.4 for both UWC versions. Within this figure, we alter the colour of the data points corresponding to minimum costs at a given $C$ to black to clearly indicate which $k_i$ they belong to. Surprisingly, we observe drastically different relationships between the costs and $C$. For UWC version 1 we observe a monotonic increasing relationship (similar to FB, which is omitted), while UWC version 2 experiences a large spike before eventually decreasing to the same limiting values.

After investigating the components of the UWC version 2 cost functions, this was found to be largely driven by the expected time waiting in system from both classes, which we plot in Figure 6.5. The reasoning behind this is as follows. For $C = 2$, the probability of being in the first service phase when the server initiates a level-$C$ busy period is small, but increases greatly with $C$. That is, the queue length is unlikely to increase to large numbers unless the server is stuck in an exceptionally long service time. Given such a $\hat{\underline{\beta}}_i^*$, we expect a large value

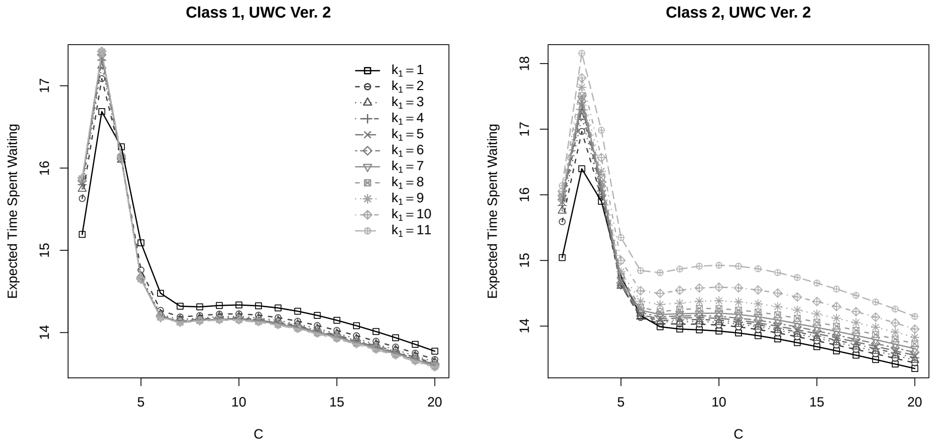

Figure 6.5 (a)

Figure 6.5 (b)

Figure 6.5: Plots of $\mathrm{E}[W_i^{\#}]$, $i = 1, 2$, at $C = 2, 3, \ldots, 20$ and $k_1 = 1, 2, \ldots, 11$, for the UWC version 2 model at $\alpha = 0.025$

of $N^*_{C,i}$ due to the excess of customers who arrive during this service, resulting in a larger UWC probability. However, this same probability is used after every observed departure, no matter if the current service time is quick or slow. Since a given service time has a 99.9% chance to be quicker, it will not in fact take very long to clear the queue after the long service completes. Therefore, the use of a larger UWC probability on subsequent services results in a much larger level-$C$ busy period than intended.

This increase in the mean level-$C$ busy period simultaneously shifts more probability mass to the buffer while making the server take longer to stop serving the class opposite the target class-$i$ customer. We then observe the mean time spent waiting in the queues overshoot the true values, rather than gradually converge to it from below like the FB model. As $C$ is increased further, less probability mass is available to be shifted to level $C$, so CTMC interactions with the UWC probabilities are less frequent and hence the inaccurate delays are experienced less often. Additionally, the total force of reneging by observed waiting customers is larger for a larger $C$, resulting in smaller UWC probabilities, naturally reducing this effect.

In contrast, we have seen UWC version 1 perform relatively well for $\mathrm{H}_2$ distributions, where service phase transitions are rare. For these particular distributions with mixing weights of 99.9% and 0.1%, they are even more so (after the initial long service time which often initiates a level-$C$ busy period). The use of phase-dependent UWC probabilities avoids the error resulting in extended level-$C$ busy periods, and so it behaves as we would expect the UWC approximation to. That is, the expected values of the time spent waiting converge to their true IB model values faster, and hence stabilize at smaller $C$. This reduces the value of $C$ after which we would no longer expect to observe more fluctuations in optimal $k_1$, as we observed in Figure 6.3. We therefore advise discretion on which UWC version to apply based on the special structures of selected service time distributions, as it is clear that UWC version 2 is not in fact strictly better than version 1.

# Bibliography

[1] Abboud, N.E. (1996). The Markovian two-echelon repairable item provisioning problem. *Journal of the Operational Research Society*, 47(2), 284-296.

[2] Adan, I.J.B.F., Economou, A., & Kapodistria, S. (2009). Synchronized reneging in queueing systems with vacations. *Queueing Systems*, 62(1-2), 1-33.

[3] Alfa, A.S., & Castro, I.T. (2002). Discrete time analysis of a repairable machine. *Journal of Applied Probability*, 39(3), 503-516.

[4] Altman, E., & Yechiali, U. (2006). Analysis of customers impatience in queues with server vacations. *Queueing Systems*, 52(4), 261-279.

[5] Artalejo, J.R., Chakravarthy, S.R., & Lopez-Herrero, M.J. (2007). The busy period and the waiting time analysis of a MAP/M/c queue with finite retrial group. *Stochastic Analysis and Applications*, 25(2), 445-469.

[6] Asmussen, S. (2000). Matrix-analytic models and their analysis. *Scandinavian Journal of Statistics*, 27(2), 193-226.

[7] Asmussen, S., Nerman, O., & Olsson, M. (1996). Fitting phase-type distributions via the EM algorithm. *Scandinavian Journal of Statistics*, 23(4), 419-441.

[8] Avrachenkov, K., Perel, E., & Yechiali, U. (2016) Finite-buffer polling systems with threshold-based switching policy. *TOP*, 24(3), 541-571.

[9] Avram, F., & Gómez-Corral, A. (2006) On the optimal control of a two-queue polling model. *Operations research letters*, 34(3), 339-348.

[10] Blanc, J.P.C. (1987). On a numerical method for calculating state probabilities for queueing systems with more than one waiting line. *Journal of Computational and Applied Mathematics*, 20, 119-125.

[11] Blanc, J.P.C. (1990). A numerical approach to cyclic-service queueing models. *Queueing Systems*, 6(1), 173-188.

[12] Blanc, J.P.C. (1991). The power-series algorithm applied to cyclic polling systems. *Communications in statistics. Stochastic models*, 7(4), 527-545.

[13] Blanc, J.P.C. (1992). An algorithmic solution of polling models with limited service disciplines. *IEEE Transactions on Communications*, 40(7), 1152-1155.

[14] Blanc, J.P.C., & van der Mei, R.D. (1995). Optimization of polling systems with Bernoulli schedules. *Performance Evaluation*, 22(2), 139-158.

[15] Boon, M.A.A. (2011). *Polling models: from theory to traffic intersections.* Doctoral dissertation, Eindhoven: Technische Universiteit Eindhoven, 190 pages.

[16] Boon, M.A.A., van der Mei, R.D., & Winands, E.M.M. (2011). Applications of polling systems. *Surveys in Operations Research and Management Science*, 16(2), 67-82.

[17] Borst, S.C., Boxma, O.J., & Levy, H. (1995). The use of service limits for efficient operation of multistation single-medium communication systems. *IEEE/ACM Transactions on Networking (TON)*, 3(5), 602-612.

[18] Boxma, O.J. (1989). Workloads and waiting times in single-server systems with multiple customer classes. *Queueing Systems*, 5(1), 185-214.

[19] Boxma, O.J., & Groenendijk, W.P. (1987). Pseudo-conservation laws in cyclic-service systems. *Journal of Applied Probability*, 24(4), 949-964.

[20] Boxma, O.J., & Groenendijk, W.P. (1988). Waiting times in discrete-time cyclic-service systems. *IEEE Transactions on Communications*, 36(2), 164-170.

[21] Boxma, O.J., Koole, G.M., & Mitrani, I. (1995) Polling models with threshold switching. In: Baccelli F, Jean-Mario A, Mitrani I (eds) Quantitative Methods in Parallel Systems, Esprit basic research series. Springer, Berlin, Heidelberg.

[22] Boxma, O.J., & Waal, P.R. (1993). *Multiserver queues with impatient customers.* Centrum voor Wiskunde en Informatica, Department of Operations Research, Statistics, and System Theory.

[23] Bright, L., & Taylor, P.G. (1995). Calculating the equilibrium distribution in level dependent quasi-birth-and-death processes. *Stochastic Models*, 11(3), 497-525.

[24] Browne, S., & Yechiali, U. (1989). Dynamic priority rules for cyclic-type queues. *Advances in Applied Probability*, 21(2), 432-450.

[25] Buyukkaramikli, N.C., van Ooijen, H.P., & Bertrand, J.W.M. (2015). Integrating inventory control and capacity management at a maintenance service provider. *Annals of Operations Research*, 231(1), 185-206.

[26] Buzen, J.P. (1973). Computational algorithms for closed queueing networks with exponential servers. *Communications of the ACM*, 16(9), 527-531.

[27] Chakravarthy, S.R. (2012). Maintenance of a deteriorating single server system with Markovian arrivals and random shocks. *European Journal of Operational Research*, 222(3), 508-522.

[28] Chang, W. & Down, D.G. (2002). Exact asympototics for $k_i$-limited exponential polling models. *Queueing Systems*, 42(4), 401-419.

[29] Ciardo, G., & Smirni, E. (1999). ETAQA: an efficient technique for the analysis of QBD-processes by aggregation. *Performance Evaluation*, 36, 71-93.

[30] Diamond, J.E., & Alfa, A.S. (1999). Matrix analytic methods for a multi-server retrial queue with buffer. *TOP*, 7(2), 249-266.

[31] Drekic, S., Stanford, D.A., Woolford, D.G., & McAlister, V.C. (2015). A model for deceased-donor transplant queue waiting times. *Queueing Systems*, 79(1), 87-115.

[32] Drekic, S., & Woolford, D.G. (2005). A preemptive priority queue with balking. *European Journal of Operational Research*, 164(2), 387-401.

[33] Fuhrmann, S.W. (1992). A decomposition result for a class of polling models. *Queueing Systems*, 11(1), 109-120.

[34] Fuhrmann, S.W., & Cooper, R.B. (1985). Stochastic decompositions in the M/G/1 queue with generalized vacations. *Operations Research*, 33(5), 1117-1129.

[35] Fuhrmann, S.W., & Wang, Y.T. (1988). Analysis of cyclic service systems with limited service: bounds and approximations. *Performance Evaluation*, 9(1), 35-54.

[36] Gaver, D.P., Jacobs, P.A., & Latouche, G. (1984). Finite birth-and-death models in randomly changing environments. *Advances in Applied Probability*, 16(4), 715-731.

[37] Gertsbakh, I. (1984). The shorter queue problem: A numerical study using the matrix-geometric solution. *European Journal of Operational Research*, 15(3), 374-381.

[38] Gordon, W.J., & Newell, G.F. (1967). Closed queuing systems with exponential servers. *Operations Research*, 15(2), 254-265.

[39] Granville, K., & Drekic, S. (2018), On a 2-class polling model with reneging and $k_i$-limited service. *Annals of Operations Research*, 274(1), 267-290. doi:10.1007/s10479-018-2915-y.

[40] Granville, K., & Drekic, S. (2018) A 2-class maintenance model with a finite population and competing exponential failure rates. *Queueing Models and Service Management*, 1(1), 141-176.

[41] Granville, K., & Drekic, S. (2019). A 2-class maintenance model with dynamic server behavior. *TOP*, 1-63. doi:10.1007/s11750-019-00509-1

[42] Gross, D., Miller, D.R., & Soland, R.M. (1983). A closed queueing network model for multi-echelon repairable item provisioning. *AIIE Transactions*, 15(4), 344-352.

[43] He, Q.M. (2014) *Fundamentals of matrix-analytic methods, vol. 365.* Springer, New York.

[44] Heindl, A., Zhang, Q., & Smirni, E. (2004). ETAQA truncation models for the MAP/MAP/1 departure process. In *Proceedings of First International Conference on the Quantitative Evaluation of Systems*, 100-109.

[45] Hsu, L.F. (1999). Simultaneous determination of preventive maintenance and replacement policies in a queue-like production system with minimal repair. *Reliability Engineering & System Safety*, 63(2), 161-167.

[46] Igaki, N. (1992). Exponential two server queue with N-policy and general vacations. *Queueing Systems*, 10(4), 279-294.

[47] Iravani, S.M., & Kolfal, B. (2005) When does the $c\mu$ rule apply to finite-population queueing systems?. *Operations Research Letters*, 33(3), 301-304.

[48] Iravani, S.M., Krishnamurthy, V., & Chao, G. H. (2007) Optimal server scheduling in nonpreemptive finite-population queueing systems. *Queueing Systems*, 55(2), 95-105.

[49] Jung, W.Y., & Un, C.K. (1994). Analysis of a finite-buffer polling system with exhaustive service based on virtual buffering. *IEEE Transactions on Communications*, 42(12), 3144-3149.

[50] Keilson, J., & Servi, L.D. (1986). Oscillating random walk models for GI/G/1 vacation systems with Bernoulli schedules. *Journal of Applied Probability*, 23(3), 790-802.

[51] Kendall, D.G. (1953). Stochastic processes occurring in the theory of queues and their analysis by the method of the imbedded Markov chain. *Annals of Mathematical Statistics*, 338-354.

[52] Kim, J., & Kim, B. (2013). Waiting time distribution in an M/PH/1 retrial queue. Performance Evaluation, 70(4), 286-299.

[53] Kim, S.K., & Dshalalow, J.H. (2003). A versatile stochastic maintenance model with reserve and super-reserve machines. *Methodology and Computing in Applied Probability*, 5(1), 59-84.

[54] Kim, W.B., & Koenigsberg, E. (1987). The efficiency of two groups of N machines served by a single robot. *Journal of the Operational Research Society*, 523-538.

[55] Kleinrock, L. (1965). A conservation law for a wide class of queueing disciplines. *Naval Research Logistics (NRL)*, 12(2), 181-192.

[56] Koenigsberg, E., & Mamer, J. (1982). The analysis of production systems. *The International Journal of Production Research*, 20(1), 1-16.

[57] Krishnamoorthy, A., Babu, S., & Narayanan, V.C. (2009). The MAP/(PH/PH)/1 queue with self-generation of priorities and non-preemptive service. *European Journal of Operational Research*, 195(1), 174-185.

[58] Lakatos, L., Szeidl, L., & Telek, M. (2012) *Introduction to queueing systems with telecommunication applications.* Springer Science & Business Media, Berlin.

[59] Lee, D.S. (1996). A two-queue model with exhaustive and limited service disciplines. *Stochastic Models*, 12(2), 285-305.

[60] Lee, D.S., & Sengupta, B. (1993) Queueing analysis of a threshold based priority scheme for ATM networks. *IEEE/ACM Transactions on Networking (TON)*, 1(6), 709-717.

[61] Levy, H., & Sidi, M. (1990). Polling systems: applications, modeling, and optimization. *IEEE Transactions on Communications*, 38(10), 1750-1760.

[62] Liang, W.K., Balcıoğlu, B., & Svaluto, R. (2013) Scheduling policies for a repair shop problem. *Annals of Operations Research*, 211(1), 273-288.

[63] Lin, C., Madu, C.N., & Kuei, C.H. (1994) A closed queuing maintenance network for a flexible manufacturing system. *Microelectronics Reliability*, 34(11), 1733-1744.

[64] Little, J.D. (1961). A proof for the queuing formula: $L = \lambda W$. *Operations Research*, 9(3), 383-387.

[65] Mack, C. (1957). The efficiency of N machines uni-directionally patrolled by one operative when walking time is constant and repair times are variable. *Journal of the Royal Statistical Society. Series B (Methodological)*, 173-178.

[66] Mack, C., Murphy, T., & Webb, N.L. (1957). The efficiency of N machines uni-directionally patrolled by one operative when walking time and repair times are constants. *Journal of the Royal Statistical Society. Series B (Methodological)*, 166-172.

[67] Madu, C.N. (1988). A closed queueing maintenance network with two repair centres. *Journal of the Operational Research society*, 959-967.

[68] Meilijson, I., & Yechiali, U. (1977). On optimal right-of-way policies at a single-server station when insertion of idle times is permitted. *Stochastic Processes and Their Applications*, 6(1), 25-32.

[69] Neuts, M.F. (1975). Computational uses of the method of phases in the theory of queues. *Computers & Mathematics with Applications*, 1(2), 151-166.

[70] Neuts, M.F. (1981). *Matrix-geometric solutions in stochastic models: an algorithmic approach.* Courier Corporation.

[71] Neuts, M.F., Pérez-Ocón, R., & Torres-Castro, I. (2000). Repairable models with operating and repair times governed by phase type distributions. *Advances in Applied Probability*, 32(2), 468-479.

[72] Ozawa, T. (1990). Alternating service queues with mixed exhaustive and K-limited services. *Performance Evaluation*, 11(3), 165-175.

[73] Palm, C. (1953). Methods of judging the annoyance caused by congestion. *Tele, 4*(189208), 4-5.

[74] Perel, E., & Yechiali, U. (2017) Two-queue polling systems with switching policy based on the queue that is not being served. *Stochastic Models*, 33(3), 1-21.

[75] Pérez-Ocón, R. & Montoro-Cazorla, D. (2004). Transient analysis of a repairable system, using phase-type distributions and geometric processes. *IEEE Transactions on Reliability*, 53(2), 185-192.

[76] Perry, D., & Posner, M.J. (2000). A correlated M/G/1-type queue with randomized server repair and maintenance modes. *Operations Research Letters*, 26(3), 137-147.

[77] Peschansky, A.I., & Kovalenko, A.I. (2016). On a strategy for the maintenance of an unreliable channel of a one-server loss queue. *Automatic Control and Computer Sciences*, 50(6), 397-407.

[78] Ramaswamy, R., & Servi, L.D. (1988). The busy period of the $M/G/1$ vacation model with a Bernoulli schedule. *Communications in Statistics. Stochastic Models*, 4(3), 507-521.

[79] Resing, J.A.C. (1993). Polling systems and multitype branching processes. *Queueing Systems*, 13(4), 409-426.

[80] Righter, R. (2002). Optimal maintenance and operation of a system with backup components. *Probability in the Engineering and Informational Sciences*, 16(3), 339-349.

[81] Riska, A., & Smirni, E. (2002). Exact aggregate solutions for M/G/1-type Markov processes. In *ACM SIGMETRICS Performance Evaluation Review*, 30(1), 86-96.

[82] Ross, S.M. (2014) *Introduction to probability models.* Academic press, San Diego.

[83] Sakuma, Y., & Takine, T. (2017). Multi-class M/PH/1 queues with deterministic impatience times. *Stochastic Models*, 33(1), 1-29.

[84] Schrage, L. (1968). Letter to the editor - a proof of the optimality of the shortest remaining processing time discipline. *Operations Research*, 16(3), 687-690.

[85] Servi, L.D. (1986). Average delay approximation of M/G/1 cyclic service queues with Bernoulli schedules. *IEEE Journal on Selected Areas in Communications*, 4(6), 813-822.

[86] Shin, Y.W., & Choo, T.S. (2009). M/M/s queue with impatient customers and retrials. *Applied Mathematical Modelling*, 33(6), 2596-2606.

[87] Stern, T. (1979). Approximations of queue dynamics and their application to adaptive routing in computer communication networks. *IEEE Transactions on Communications*, 27(9), 1331-1335.

[88] Syski, R. (1992) *Passage times for Markov chains.* IOS Press, Amsterdam.

[89] Takagi, H. (1988). Queuing analysis of polling models. *ACM Computing Surveys (CSUR)*, 20(1), 5-28.

[90] Towsley, D. (1980). Queuing network models with state-dependent routing. *Journal of the ACM (JACM)*, 27(2), 323-337.

[91] van Mieghem, J.A. (1995) Dynamic scheduling with convex delay costs: The generalized $c\mu$ rule. *Annals of Applied Probability*, 5(3), 809-833.

[92] van Vuuren, M., & Winands, E.M.M. (2007). Iterative approximation of $k$-limited polling systems. *Queueing Systems*, 55(3), 161-178.

[93] Vishnevskii, V.M., & Semenova, O.V. (2006). Mathematical methods to study the polling systems. *Automation and Remote Control*, 67(2), 173-220.

[94] Wang, J., Baron, O., & Scheller-Wolf, A. (2015). M/M/c queue with two priority classes. *Operations Research*, 63(3), 733-749.

[95] Weststrate, J.A., & van der Mei, R.D. (1994) Waiting times in a two-queue model with exhaustive and Bernoulli service. *Zeitschrift für Operations Research*, 40(3), 289-303.

[96] Winands, E.M.M., Adan, I.J.B.F., & van Houtum, G.J. (2006). Mean value analysis for polling systems. *Queueing Systems*, 54(1), 35-44.

[97] Winands, E.M.M., Adan, I.J.B.F., van Houtum, G.J., & Down, D.G. (2009). A state-dependent polling model with k-limited service. *Probability in the Engineering and Informational Sciences*, 23(2), 385-408.

[98] Wolff, R.W. (1982). Poisson arrivals see time averages. *Operations Research*, 30(2), 223-231.

[99] Yang, W. S., Lim, D. E., & Chae, K. C. (2009). Maintenance of deteriorating single server queues with random shocks. *Computers & Industrial Engineering*, 57(4), 1404-1406.

[100] Yechiali, U. (2007). Queues with system disasters and impatient customers when system is down. *Queueing Systems*, 56(3), 195-202.

# Appendix A

## A.1 Derivation of the Kolmogorov Backward Equations

We will derive the Kolmogorov Backward equations for a CTMC whose state space $\mathcal{S}$ may have countable-many states. We begin by considering how to calculate the transition probability function, $P_{i,j}(t)$. Considering if the CTMC has left state $i$ by time $t$, from the law of total probability we obtain

$$P(X(t)=j|X(0)=i) = P(X(t)=j, T_i > t|X(0)=i) + P(X(t)=j, T_i \leq t|X(0)=i).$$

Clearly, if $T_i > t$, then the CTMC may be in state $j$ iff $j = i$, and so

$$P(X(t)=j, T_i > t|X(0)=i) = P(X(t)=j|T_i > t, X(0)=i)P(T_i > t|X(0)=i) = \delta_{i,j}e^{-v_i t}.$$

If we now assume that $T_i \leq t$, then considering what state $k \neq i$ the CTMC transitions to at $T_i$, we get

$$P(X(t)=j, T_i \leq t|X(0)=i) = \sum_{\substack{k\in\mathcal{S}\\ k\neq i}} P(X(t)=j, T_i \leq t, X(T_i)=k|X(0)=i).$$

Conditioning on the value of $T_i$,

$$\begin{aligned}
&P(X(t)=j, T_i \leq t, X(T_i)=k|X(0)=i)\\
&= \int_0^\infty P(X(t)=j, T_i \leq t, X(T_i)=k|X(0)=i, T_i=s)f_{T_i}(s)ds\\
&= \int_0^t P(X(t)=j, T_i \leq t, X(T_i)=k|X(0)=i, T_i=s)f_{T_i}(s)ds.
\end{aligned}$$

From the definition of conditional probability, it holds that

$$\begin{aligned}
&P(X(t)=j, T_i \leq t, X(T_i)=k|X(0)=i, T_i=s)\\
&= P(X(t)=j|X(s)=k, X(u)=i, 0 \leq u < s)P(X(s)=k|X(0)=i, T_i=s),
\end{aligned}$$

where we can note that $P(X(s)=k|X(0)=i, T_i=s) = p_{i,k} = q_{i,k}/v_i$. Thus, by the Markov property and the stationary assumption of CTMCs,

$$\begin{aligned}
P(X(t)=j, T_i \leq t|X(0)=i) &= \sum_{\substack{k\in\mathcal{S}\\ k\neq i}} \int_0^t P(X(t)=j|X(s)=k)\frac{q_{i,k}}{v_i}v_i e^{-v_i s}ds\\
&= \sum_{\substack{k\in\mathcal{S}\\ k\neq i}} \int_0^t q_{i,k}e^{-v_i s}P_{k,j}(t-s)ds.
\end{aligned}$$

Now, note that since $q_{i,k}$, $e^{-v_i s}$, and $P_{k,j}(t-s)$ are non-negative for all $k \in \mathcal{S}$, $k \neq i$, as a consequence of the monotone convergence theorem we may interchange the order of summation and integration and obtain

$$\sum_{\substack{k \in \mathcal{S} \\ k \neq i}} \int_0^t q_{i,k} e^{-v_i s} P_{k,j}(t-s) ds = \int_0^t \sum_{\substack{k \in \mathcal{S} \\ k \neq i}} q_{i,k} e^{-v_i s} P_{k,j}(t-s) ds.$$

Thus,

$$\begin{aligned} P_{i,j}(t) &= P(X(t) = j, T_i > t | X(0) = i) + P(X(t) = j, T_i \leq t | X(0) = i) \\ &= \delta_{i,j} e^{-v_i t} + \int_0^t \sum_{\substack{k \in \mathcal{S} \\ k \neq i}} q_{i,k} e^{-v_i s} P_{k,j}(t-s) ds, \end{aligned}$$

and after multiplying both sides by $e^{v_i t}$, we obtain

$$e^{v_i t} P_{i,j}(t) = \delta_{i,j} + \int_0^t \sum_{\substack{k \in \mathcal{S} \\ k \neq i}} q_{i,k} e^{v_i (t-s)} P_{k,j}(t-s) ds = \delta_{i,j} + \int_0^t \sum_{\substack{k \in \mathcal{S} \\ k \neq i}} q_{i,k} e^{v_i u} P_{k,j}(u) du.$$

Taking the derivative with respect to $t$ and applying the Leibniz integral rule to evaluate the derivative of the right-hand side, we get

$$v_i e^{v_i t} P_{i,j}(t) + e^{v_i t} P'_{i,j}(t) = \sum_{\substack{k \in \mathcal{S} \\ k \neq i}} q_{i,k} e^{v_i t} P_{k,j}(t).$$

Finally, after multiplying both sides by $e^{-v_i t}$, we are able to recover the Kolmogorov Backward equations:

$$P'_{i,j}(t) = \sum_{\substack{k \in \mathcal{S} \\ k \neq i}} q_{i,k} P_{k,j}(t) - v_i P_{i,j}(t).$$

□

## A.2 Proof of Corollary 1.1

We will now prove the three necessary claims required for Corollary 1.1 to hold true. Recall that we need only consider non-absorbing states $i \in S$, where $v_i > 0$ and there exists at least one $j \in S$ such that $q_{i,j} > 0$. First, let us consider the distribution of a sojourn time in state $i$. Let $Y_{i,j} \sim \text{Exp}(q_{i,j})$, $j \neq i$, $j \in S$, be independent exponentially distributed random variables such that if $q_{i,j} = 0$ then we let $Y_{i,j} = \infty$ with probability 1.

As we have claimed that the CTMC will leave state $i$ after observing the first timer completion, a sojourn time $T_i$ is simply equal in distribution to the minimum of $Y_{i,j}$, $j \neq i$, $j \in S$.

Therefore, for any $t > 0$,

$$\begin{aligned}
P(T_i > t) &= P(\min\{Y_{i,j}, j \neq i\} > t) \\
&= P(Y_{i,j} > t, j \neq i) \\
&= \prod_{j \neq i} P(Y_{i,j} > t) && \text{Independence} \\
&= \prod_{j \neq i} e^{-q_{i,j} t} \\
&= e^{-\left(\sum_{j \neq i} q_{i,j}\right) t} \\
&= e^{-v_i t},
\end{aligned}$$

implying that $T_i \sim \text{Exp}(v_i)$, as required. Note that in the above, if $q_{i,j} = 0$, then $P(Y_{i,j} > t) = e^0 = 1$.

Next, we consider the probability that when the CTMC leaves state $i$, it transitions to state $j \neq i$, $p_{i,j}$. If $q_{i,j} = 0$, then it is trivial to confirm that $p_{i,j} = 0$ (since $Y_{i,j} = \infty$), so let us suppose that $q_{i,j} > 0$. We will observe the CTMC transition to $j$ iff $Y_{i,j} = \min\{Y_{i,k}, k \neq i\}$, or equivalently, $Y_{i,j} < \min\{Y_{i,k}, k \neq i, k \neq j\}$. If we let $Y^*_{i,j} = \min\{Y_{i,k}, k \neq i, j\}$, by the above, it follows that $Y^*_{i,j} \sim \text{Exp}\left(\sum_{k \neq i,j} q_{i,k} = v_i - q_{i,j}\right)$.

If $v_i = q_{i,j}$, then $Y^*_{i,j} = \infty$ with probability 1, and

$$P(Y_{i,j} < Y^*_{i,j}) = P(Y_{i,j} < \infty) = 1 = p_{i,j},$$

as required. Suppose now that $v_i > q_{i,j}$. Recalling that $q_{i,j} = v_i p_{i,j}$, it follows that

$$\begin{aligned}
P(Y_{i,j} < Y^*_{i,j}) &= \int_0^\infty P(Y_{i,j} < Y^*_{i,j} | Y_{i,j} = y) f_{Y_{i,j}}(y) dy \\
&= \int_0^\infty P(Y^*_{i,j} > y) f_{Y_{i,j}}(y) dy && \text{Independence} \\
&= \int_0^\infty e^{-(v_i - q_{i,j})y} q_{i,j} e^{-q_{i,j} y} dy \\
&= p_{i,j} \int_0^\infty v_i e^{-v_i y} dy = p_{i,j},
\end{aligned}$$

as required.

Let $A_{i,j} = \{Y_{i,j} < Y^*_{i,j}\}$ denote the event that after a visit to state $i$, the CTMC next transitions to state $j \neq i$ (such that $P(A_{i,j}) = p_{i,j}$). Supposing that $q_{i,j} > 0$ (i.e., $p_{i,j} > 0$) and applying the definition of conditional probability, we have

$$P(T_i > t, A_{i,j}) = P(T_i > t | A_{i,j}) P(A_{i,j}),$$

and

$$\begin{aligned}
P(T_i > t | A_{i,j}) &= P(Y_{i,j} > t | Y_{i,j} < Y^*_{i,j}) \\
&= \frac{P(Y_{i,j} > t, Y_{i,j} < Y^*_{i,j})}{P(Y_{i,j} < Y^*_{i,j})} \\
&= \frac{1}{p_{i,j}} \int_0^\infty P(Y_{i,j} > t, Y_{i,j} < Y^*_{i,j} | Y_{i,j} = y) f_{Y_{i,j}}(y) dy \\
&= \frac{1}{p_{i,j}} \int_t^\infty P(Y^*_{i,j} > y) f_{Y_{i,j}}(y) dy \qquad \text{Independence} \\
&= \frac{1}{p_{i,j}} \int_t^\infty e^{-(v_i - q_{i,j})y} q_{i,j} e^{-q_{i,j}y} dy \\
&= \int_t^\infty v_i e^{-v_i y} dy \\
&= P(T_i > t),
\end{aligned}$$

so

$$P(T_i > t, A_{i,j}) = P(T_i > t) P(A_{i,j}).$$

Lastly, if $q_{i,j} = 0$, then $P(T_i > t, A_{i,j}) = 0$ and $P(A_{i,j}) = p_{i,j} = 0$, so the above equality still holds true. Thus, we have proven that $T_i$ and $A_{i,j}$ are independent for all $j \in S$, and have now confirmed that the claim in Corollary 1.1 is true.

□

## A.3 Proof of Theorem 3.1

We begin by remarking that the infinitesimal generator subblocks $Q^{[C,f]}_{0,1,0}$ and $(UD)^{[C,f]}_{0,0}$ both comprise $1 + s_0$ identical rows equal to $C\alpha_1 \underline{\gamma}^{[+0]}_{01} \otimes \underline{\beta}_1$ and $C\alpha_2 \underline{\gamma}^{[+0]}_{02} \otimes \underline{\beta}_2$, respectively. This implies that given a machine failure has occurred, the CTMC transitions away from any of the empty queue states $\{(0,0,0,0,0,0)\} \cup \{(0,0,5,y,0,0), y = 1,2,\ldots,s_0\}$ in an identical fashion. This observation immediately follows from our assumption that interrupting and switching away from a class-$i$ switch-in is treated the same as the server switching away from class $i$ itself.

We now consider the differences between a system containing $k$ machines where $[C,f] = [k,0]$ or $[C,f] = [k-1,1]$. The two systems will act identically, in terms of infinitesimal generator construction, with the exception of the rows for states where all $k$ machines are functional (the first system puts the $k^{\text{th}}$ machine to use, while the second stores it in the maintenance float). In either case, the total time spent visiting any combination of the empty queue states between the previous service completion and the next observed failure will have an exponential distribution with rate $C\alpha$ (i.e., $\text{Exp}(C\alpha)$). Hence, we may adjust the CTMCs and consolidate the empty queue states into a single state $(0,0)$ with steady-state probability $\pi^{[C,f]}_{0,0} = \underline{\pi}^{[C,f]}_{0,0} \underline{e}'_{1+s_0}$, such that we do not track the potential phase-type class-0 switch-in time and the sojourn time in this state is simply the time until the next machine failure. Note that this consolidation will not affect the other steady-state probabilities due to the identical rows of $Q^{[C,f]}_{0,1,0}$ and $(UD)^{[C,f]}_{0,0}$ which each are now just present once, corresponding to transitions out of state $(0,0)$. Thus, we have at steady state

$$\begin{aligned}
\mathrm{E}[N_{\mathrm{W}}^{[C,f]}] &= \mathrm{E}[\min\{C, C+f-X_1^{[C,f]}-X_2^{[C,f]}\}] \\
&= \sum_{m,n,l,y,y_1,y_2} \min\{C, C+f-m-n\}\pi^{[C,f]}_{m,n,l,y,y_1,y_2} \\
&= C\left(\pi^{[C,f]}_{0,0,0,0,0,0} + \sum_{y=1}^{s_0}\pi^{[C,f]}_{0,0,5,y,0,0}\right) \\
&\quad + \sum_{m+n\neq 0}\sum_{l,y,y_1,y_2} \min\{C, C+f-m-n\}\pi^{[C,f]}_{m,n,l,y,y_1,y_2} \\
&= C\pi^{[C,f]}_{0,0} + \sum_{m+n\neq 0}\sum_{l,y,y_1,y_2} \min\{C, C+f-m-n\}\pi^{[C,f]}_{m,n,l,y,y_1,y_2}. \qquad \text{(A.1)}
\end{aligned}$$

That is, the expected number of working machines will be the same in the original CTMCs and the corresponding adjusted CTMCs with the consolidated empty queue state.

Let $\psi^{[C,f]}_{0,0}$ and $\psi^{[C,f]}_{m,n,l,y,y_1,y_2}$ denote the steady-state probabilities of the embedded DTMC (e.g., Syski [88], p. 14), describing an adjusted CTMC with a given $[C, f]$. As the generators for $[k, 0]$ and $[k-1, 1]$ are now identical outside of the first rows for state $(0, 0)$, which for $[k, 0]$ is

$$\left[\begin{array}{cccccccccc} -k\alpha & k\alpha_2\underline{\gamma}_{02}^{[+0]}\otimes\underline{\beta}_2 & \underline{0} & \cdots & \underline{0} & k\alpha_1\underline{\gamma}_{01}^{[+0]}\otimes\underline{\beta}_1 & \underline{0} & \cdots & \underline{0} \end{array}\right]$$

and for $[k-1, 1]$ is

$$\left[\begin{array}{cccccccccc} -(k-1)\alpha & (k-1)\alpha_2\underline{\gamma}_{02}^{[+0]}\otimes\underline{\beta}_2 & \underline{0} & \cdots & \underline{0} & (k-1)\alpha_1\underline{\gamma}_{01}^{[+0]}\otimes\underline{\beta}_1 & \underline{0} & \cdots & \underline{0} \end{array}\right],$$

it is clear that while the steady-state probabilities for the CTMCs differ, it holds that $\psi^{[k,0]}_{0,0} = \psi^{[k-1,1]}_{0,0}$ and $\psi^{[k,0]}_{m,n,l,y,y_1,y_2} = \psi^{[k-1,1]}_{m,n,l,y,y_1,y_2}$.

It is known from the theory of semi-Markov processes (e.g., Ross [82], p. 445) that if the long-run proportion of transitions by a semi-Markov process into state $i$ is $\pi_i$ (i.e., the steady-state probability of the embedded DTMC being in state $i$) and the amount of time spent in state $i$ before transitioning away has mean $\mu_i$, then the long-run proportion of time that the semi-Markov process is in state $i$ is

$$\frac{\pi_i\mu_i}{\sum_{j=1}^{N}\pi_j\mu_j}, \qquad \text{(A.2)}$$

where $N$ is the total number of states. Since we are considering CTMCs, the time spent in a state is exponentially distributed with a mean equal to the negative inverse of that state's corresponding main diagonal element from the infinitesimal generator. Let $\mu^{[k,0]}_{m,n,l,y,y_1,y_2} = \mu^{[k-1,1]}_{m,n,l,y,y_1,y_2}$ denote the mean time spent in a visit to state $(m, n, l, y, y_1, y_2)$, and $\mu^{[k,0]}_{0,0} = \frac{1}{k\alpha}$ and $\mu^{[k-1,1]}_{0,0} = \frac{1}{(k-1)\alpha}$ be the mean times spent in visits to the empty queue state in either adjusted CTMC. We then have

$$\pi^{[C,f]}_{m,n,l,y,y_1,y_2} = \frac{\psi^{[C,f]}_{m,n,l,y,y_1,y_2}\mu^{[C,f]}_{m,n,l,y,y_1,y_2}}{\psi^{[C,f]}_{0,0}\mu^{[C,f]}_{0,0} + \sum_{x_1+x_2\neq 0}\sum_{w,z,z_1,z_2}\psi^{[C,f]}_{x_1,x_2,w,z,z_1,z_2}\mu^{[C,f]}_{x_1,x_2,w,z,z_1,z_2}}$$

and

$$\pi_{0,0}^{[C,f]} = \frac{\psi_{0,0}^{[C,f]}\mu_{0,0}^{[C,f]}}{\psi_{0,0}^{[C,f]}\mu_{0,0}^{[C,f]} + \sum_{x_1+x_2\neq 0}\sum_{w,z,z_1,z_2}\psi_{x_1,x_2,w,z,z_1,z_2}^{[C,f]}\mu_{x_1,x_2,w,z,z_1,z_2}^{[C,f]}}.$$

Let

$$D^{[C,f]} = \sum_{x_1+x_2\neq 0}\sum_{w,z,z_1,z_2}\psi_{x_1,x_2,w,z,z_1,z_2}^{[C,f]}\mu_{x_1,x_2,w,z,z_1,z_2}^{[C,f]},$$

which we know satisfies $D^{[k,0]} = D^{[k-1,1]}$. It now follows that

$$\begin{aligned}\pi_{m,n,l,y,y_1,y_2}^{[k,0]} &= \frac{\psi_{m,n,l,y,y_1,y_2}^{[k,0]}\mu_{m,n,l,y,y_1,y_2}^{[k,0]}}{\psi_{0,0}^{[k,0]}\mu_{0,0}^{[k,0]} + D^{[k,0]}} \\ &= \frac{\psi_{m,n,l,y,y_1,y_2}^{[k-1,1]}\mu_{m,n,l,y,y_1,y_2}^{[k-1,1]}}{\psi_{0,0}^{[k-1,1]}\mu_{0,0}^{[k-1,1]} + D^{[k-1,1]}} \times \frac{\psi_{0,0}^{[k-1,1]}\mu_{0,0}^{[k-1,1]} + D^{[k-1,1]}}{\psi_{0,0}^{[k,0]}\mu_{0,0}^{[k,0]} + D^{[k,0]}} \\ &= \pi_{m,n,l,y,y_1,y_2}^{[k-1,1]}c_k, \qquad \text{(A.3)}\end{aligned}$$

where

$$c_k = \frac{\psi_{0,0}^{[k-1,0]}\mu_{0,0}^{[k-1,1]} + D^{[k-1,1]}}{\psi_{0,0}^{[k,0]}\mu_{0,0}^{[k,0]} + D^{[k,0]}} = \frac{\frac{1}{(k-1)\alpha}\psi_{0,0}^{[k,0]} + D^{[k,0]}}{\frac{1}{k\alpha}\psi_{0,0}^{[k,0]} + D^{[k,0]}} > 1. \qquad \text{(A.4)}$$

Similarly,

$$\begin{aligned}\pi_{0,0}^{[k,0]} &= \frac{\psi_{0,0}^{[k,0]}\mu_{0,0}^{[k,0]}}{\psi_{0,0}^{[k,0]}\mu_{0,0}^{[k,0]} + D^{[k,0]}} \\ &= \frac{\psi_{0,0}^{[k-1,1]}\mu_{0,0}^{[k-1,1]}\left(\frac{k-1}{k}\right)}{\psi_{0,0}^{[k-1,1]}\mu_{0,0}^{[k-1,1]} + D^{[k-1,1]}} \times \frac{\psi_{0,0}^{[k-1,1]}\mu_{0,0}^{[k-1,1]} + D^{[k-1,1]}}{\psi_{0,0}^{[k,0]}\mu_{0,0}^{[k,0]} + D^{[k,0]}} \\ &= \pi_{0,0}^{[k-1,1]}\left(\frac{k-1}{k}\right)c_k. \qquad \text{(A.5)}\end{aligned}$$

Note that we can find an upper bound on $c_k$. As the steady-state probabilities for both cases must respectively sum to 1, using Equations (A.3) and (A.5), it must simultaneously hold that

$$\begin{aligned}1 &= \pi_{0,0}^{[k,0]} + \sum_{x_1+x_2\neq 0}\sum_{w,z,z_1,z_2}\pi_{x_1,x_2,w,z,z_1,z_2}^{[C,f]} \\ &= \pi_{0,0}^{[k-1,1]}\left(\frac{k-1}{k}\right)c_k + c_k\sum_{x_1+x_2\neq 0}\sum_{w,z,z_1,z_2}\pi_{m,n,l,y,y_1,y_2}^{[k-1,1]}\end{aligned}$$

and

$$1 = \pi_{0,0}^{[k-1,1]} + \sum_{x_1+x_2\neq 0}\sum_{w,z,z_1,z_2}\pi_{m,n,l,y,y_1,y_2}^{[k-1,1]}.$$

Clearly, as every probability is non-negative, by Equation (A.4),

$$c_k\sum_{x_1+x_2\neq 0}\sum_{w,z,z_1,z_2}\pi_{m,n,l,y,y_1,y_2}^{[k-1,1]} > \sum_{x_1+x_2\neq 0}\sum_{w,z,z_1,z_2}\pi_{m,n,l,y,y_1,y_2}^{[k-1,1]}$$

implying that we must have

$$\pi_{0,0}^{[k-1,1]}\left(\frac{k-1}{k}\right)c_k < \pi_{0,0}^{[k-1,1]},$$

or equivalently,

$$1 < c_k < \frac{k}{k-1}.$$

Finally, using Equations (A.1) - (A.5),

$$\begin{aligned}
&\mathrm{E}[N_{\mathrm{W}}^{[k,0]}] \\
&= k\pi_{0,0}^{[k,0]} + \sum_{m+n\neq 0}\sum_{l,y,y_1,y_2} \min\{k, k+0-m-n\}\pi_{m,n,l,y,y_1,y_2}^{[k,0]} \\
&= k\pi_{0,0}^{[k,0]} + \sum_{m+n\neq 0}\sum_{l,y,y_1,y_2} (k-m-n)\pi_{m,n,l,y,y_1,y_2}^{[k,0]} \\
&= k\pi_{0,0}^{[k-1,1]}\left(\frac{k-1}{k}\right)c_k + \sum_{m+n\neq 0}\sum_{l,y,y_1,y_2} (k-m-n)\pi_{m,n,l,y,y_1,y_2}^{[k-1,1]}c_k \\
&= c_k\left((k-1)\pi_{0,0}^{[k-1,1]} + \sum_{m+n\neq 0}\sum_{l,y,y_1,y_2} \min\{k-1, k-1+1-m-n\}\pi_{m,n,l,y,y_1,y_2}^{[k-1,1]}\right) \\
&= c_k\mathrm{E}[N_{\mathrm{W}}^{[k-1,1]}] \\
&> \mathrm{E}[N_{\mathrm{W}}^{[k-1,1]}].
\end{aligned}$$

□

## A.4 Proof of Theorem 3.2

In order to consider the limit of the expected number of working machines, we need to first find an expression for $\mathrm{E}[N_{\mathrm{W}}^{[C,f]}]$. Similar to Abboud [1], we consider the number of working machines as a subsystem and apply the result of Little [64]. Recall that Little's Law states that the expected number of 'customers' in a system ($\mathrm{E}[L]$) is equal to the product of their average arrival rate ($\lambda$) and the expected amount of time that a customer spends in the system ($\mathrm{E}[W]$).

As we are treating the number of *working* machines as the subsystem, not the number of *functional* machines, it is clear that $W$ is simply the time until a working machine fails. Thus, we have $W \sim \mathrm{Exp}(\alpha)$, and so

$$\mathrm{E}[W] = \frac{1}{\alpha}, \tag{A.6}$$

which is independent of $C$, $f$, and the service policy. Next, we require the limiting aggregate rate that machines fail and are repaired, which we define as $\lambda_r^{[C,f]}$, which is the effective average 'arrival rate' of repaired machines satisfying

$$\mathrm{E}[N_{\mathrm{W}}^{[C,f]}] = \lambda_r^{[C,f]}\mathrm{E}[W] = \frac{\lambda_r^{[C,f]}}{\alpha}. \tag{A.7}$$

We cite a result from the theory of renewal reward processes (e.g., Ross [82], p. 427), describing a system which earns a reward $R_n$ after the $n^{\text{th}}$ renewal of a renewal process $\{N(t), t \geq 0\}$ with interarrival times $X_n$, $n \in \mathbb{Z}^+$, where the $R_n$'s are iid, but may depend on $X_n$. The total amount of rewards that have accumulated by time $t \geq 0$ is

$$R(t) = \sum_{n=1}^{N(t)} R_n,$$

and it is known that the long run rate at which rewards are earned is

$$\lim_{t \to \infty} \frac{R(t)}{t} = \frac{\mathrm{E}[R]}{\mathrm{E}[X]}. \tag{A.8}$$

We now define a renewal process based on our adjusted model from the proof of Theorem 3.1 with $[C, f]$ machines, such that a renewal occurs whenever the adjusted CTMC enters the empty queue state $(0, 0)$ (i.e., at time instants immediately after a repair which leaves all machines functional). At the end of each renewal, we receive a reward of 1 unit per observed service completion during that cycle. Applying Equation (A.8) to this renewal process will result in the aggregate rate at which machines are repaired. That is, if we let $\mathrm{E}[BP^{[C,f]}]$ denote the mean duration of a busy period (i.e., the time between a failure to an empty system and when the system is empty again), then

$$\lambda_r^{[C,f]} = \frac{\mathrm{E}[\text{Number of repairs in } BP^{[C,f]}]}{\mathrm{E}[\text{Time until first failure at full capacity}] + \mathrm{E}[BP^{[C,f]}]}. \tag{A.9}$$

Let $BP_{\text{ser}}^{[C,f]}$ and $BP_{\text{swi}}^{[C,f]}$ denote the time spent serving or switching during a busy period, respectively, such that $BP^{[C,f]} = BP_{\text{ser}}^{[C,f]} + BP_{\text{swi}}^{[C,f]}$. Note that regardless of order caused by a particular service policy, every machine that fails during (or initiating) the busy period must eventually be served. Since we assume that any preempted services are resumed when the server returns, no work is lost due to switch-ins. Therefore, if for example a class-2 repair time has the potential to be interrupted until some number of class-1 repairs are completed, the total expected time to repair that class-2 machine is still $-\underline{\beta}_2 B_2^{-1} \underline{e}'$. Thus, if we let $N_{BP}$ be the number of repairs in $BP^{[C,f]}$, then $BP_{\text{ser}}^{[C,f]}$ can be represented as the sum of all total service times observed during the busy period

$$BP_{\text{ser}}^{[C,f]} = \sum_{n=1}^{N_{BP}} Z_n^{\mathrm{M}},$$

where $Z_n^{\mathrm{M}}$, $n = 1, 2, \ldots$, are iid random service times which are mixtures of $\mathrm{PH}_{b_i}(\underline{\beta}_i, B_i)$ distributions, $i = 1, 2$, with weights $\alpha_1/\alpha$ and $\alpha_2/\alpha$, having mean

$$\mathrm{E}[Z^{\mathrm{M}}] = -\left(\frac{\alpha_1}{\alpha}\right) \underline{\beta}_1 B_1^{-1} \underline{e}' - \left(\frac{\alpha_2}{\alpha}\right) \underline{\beta}_2 B_2^{-1} \underline{e}'.$$

Therefore, it follows that for large $C$,

$$\mathrm{E}[BP_{\text{ser}}^{[C,f]}] = \mathrm{E}[\text{Number of repairs in } BP^{[C,f]}]\mathrm{E}[Z^{\mathrm{M}}],$$

where we remark that as $C \to \infty$, the duration of a busy period (and hence the number of machines serviced during a busy period) will only have a *very* weak dependence on an individual service time. Therefore, for large $C$, Equation (A.9) becomes

$$\lambda_r^{[C,f]} = \frac{\mathrm{E}[BP_{\text{ser}}^{[C,f]}]/\mathrm{E}[Z^{\mathrm{M}}]}{\frac{1}{C\alpha} + \mathrm{E}[BP_{\text{ser}}^{[C,f]}] + \mathrm{E}[BP_{\text{swi}}^{[C,f]}]}. \tag{A.10}$$

It should be noted that the distributions of $N_{BP}$, $BP^{[C,f]}_{\text{ser}}$, and $BP^{[C,f]}_{\text{swi}}$ (and hence $BP^{[C,f]}$) depend not only on $C$ and $f$, but also on the switch-in decision probabilities. For example, a class-1 preemptive resume priority discipline will always choose to clear out the small jobs as they arrive, which will result in those machines being able to fail again sooner than if the class-2 queue had to be emptied first, hence making it more likely that the server will need to repair more total machines during that busy period in comparison to other policies. We note however that the sole act of serving more machines during a busy period, and hence between renewals, does not necessarily mean that its resulting $\lambda^{[C,f]}_{f}$ will be smaller or larger, as it very much also depends on whether these extra switches (relative to other disciplines) cause idle periods due to non-zero switch-in times.

We now consider the first of three cases, where $\gamma^{[0]}_{ji} = 1 \; \forall \; i, j \in \{0, 1, 2\}, i \neq j$. Clearly, this implies that $\text{E}[BP^{[C,f]}_{\text{swi}}] = 0$, and Equation (A.10) simplifies to

$$\lambda^{[C,f]}_{r} = \frac{\text{E}[BP^{[C,f]}_{\text{ser}}]/\text{E}[Z^{\text{M}}]}{\frac{1}{C\alpha} + \text{E}[BP^{[C,f]}_{\text{ser}}]}. \tag{A.11}$$

Since $\text{E}[BP^{[C,f]}_{\text{ser}}] \geq \text{E}[Z^{\text{M}}] > 0 \; \forall \; C = 1, 2, \ldots$ and $\text{E}[BP^{[C,f]}_{\text{ser}}]$ is an increasing function in $C$ (as we will discuss shortly), by taking the limit of Equation (A.11), we observe that

$$\begin{aligned}
\lambda^{[\infty]}_{r} &= \lim_{C\to\infty} \lambda^{[C,f]}_{r} \\
&= \lim_{C\to\infty} \frac{\text{E}[BP^{[C,f]}_{\text{ser}}]/\text{E}[Z^{\text{M}}]}{\frac{1}{C\alpha} + \text{E}[BP^{[C,f]}_{\text{ser}}]} \\
&= \lim_{C\to\infty} \left( \frac{1}{\frac{1}{C\alpha \text{E}[BP^{[C,f]}_{\text{ser}}]} + 1} \right) \frac{1}{\text{E}[Z^{\text{M}}]} \\
&= \frac{-\alpha}{\alpha_1 \underline{\beta}_1 B_1^{-1} \underline{e}' + \alpha_2 \underline{\beta}_2 B_2^{-1} \underline{e}'}.
\end{aligned} \tag{A.12}$$

Therefore, Equation (3.11) follows immediately from Little's Law and Equations (A.6) and (A.12).

Next, suppose that only switches out of or into class 0 can have positive durations. It then follows that $\text{E}[BP^{[C,f]}_{\text{swi}}]$ is a constant with respect to $C$, and so it still holds that

$$\lambda^{[\infty]}_{r} = \lim_{C\to\infty} \left( \frac{1}{\frac{(C\alpha)^{-1} + \text{E}[BP^{[C,f]}_{\text{swi}}]}{\text{E}[BP^{[C,f]}_{\text{ser}}]} + 1} \right) \frac{1}{\text{E}[Z^{\text{M}}]} = \frac{-\alpha}{\alpha_1 \underline{\beta}_1 B_1^{-1} \underline{e}' + \alpha_2 \underline{\beta}_2 B_2^{-1} \underline{e}'},$$

resulting in the statement of Equation (3.11).

Finally, we consider the cases where positive switch-in times are observable in at least one direction between the class-1 and class-2 queues (i.e., $\gamma^{[0]}_{12}$ and/or $\gamma^{[0]}_{21}$ are less than 1). We now make the seemingly obvious claim that both $\text{E}[BP^{[C,f]}_{\text{ser}}]$ and $\text{E}[BP^{[C,f]}_{\text{swi}}]$ are increasing functions in $C$. This is intuitive, as increasing $C$ increases the probability flow, and hence the transition probabilities, for a given state to states within the CTMC corresponding to longer queue lengths. Also, increasing $C$ increases the maximum total queue lengths that if visited, represent more potential total work that must be completed before the end of the busy period than a corresponding 'full queue' state (i.e., $X_1(t) + X_2(t) = C + f$) in a maintenance system with a smaller $C$. Thus, the expected number of machine failures within a renewal period must increase with $C$, implying that $\mathrm{E}[BP^{[C,f]}_{\text{ser}}]$ is an increasing function in $C$.

If machine failures are more frequent, then it also follows that the probability of observing no arrivals to the opposite queue while emptying their current queue goes to zero as $C \to \infty$. To see this, consider the system at the start of a class-$i$ service while $X_i(t) = 1$ and $X_j(t) = 0$, $j \neq i$. If we assume that $f \geq 1$ and let $W_C \sim \mathrm{Exp}(C\alpha)$ and $Ser_i \sim \mathrm{PH}_{b_i}(\underline{\beta}_i, B_i)$ be independent random variables, then the probability of having no failures during this class-$i$ service is $P(W_C > Ser_i)$, where

$$P(W_C > Ser_i) = \int_0^\infty e^{-C\alpha t}\underline{\beta}_i \exp\{B_i t\}\underline{e}' dt = \mathrm{E}[e^{-C\alpha Ser_i}] = \widetilde{Ser}_i(C\alpha)$$

is the Laplace transform of $Ser_i$ at $C\alpha$. If instead we had $f = 0$, then $C$ would be replaced by $C - 1$ in the above equation. Applying the dominated convergence theorem, it is easy to confirm that

$$\lim_{C\to\infty} P(W_C > Ser_i) = \lim_{C\to\infty} \widetilde{Ser}_i(C\alpha) = 0.$$

Thus, as we increase $C$, it becomes more likely that there is a combination of class-1 and/or class-2 arrivals by the end of the service. If at least one failure was from class $j$, $j \neq i$, then the server will have to undergo a class-$j$ switch-in after eventually emptying the class-$i$ queue. If every failure was class $i$, then the server will have at least one more independent and probabilistically identical opportunity to observe class-$j$ failures before either switching to class $j$ or to class 0 (and ending the busy period). Thus, the expected number of transitions between queues after emptying a queue increases with $C$, which are present for every service policy. Similarly, the number of switches from positive queue lengths will be non-decreasing in $C$ due to the CTMC spending more time at higher queue lengths, as discussed previously. Therefore, we can conclude that $\mathrm{E}[BP^{[C,f]}_{\text{swi}}]$ is also an increasing function in $C$.

Now, we rewrite Equation (A.10) as

$$\lambda_r^{[C,f]} = \left(1 + \frac{1}{C\alpha \mathrm{E}[BP^{[C,f]}_{\text{ser}}]} + \frac{\mathrm{E}[BP^{[C,f]}_{\text{swi}}]}{\mathrm{E}[BP^{[C,f]}_{\text{ser}}]}\right)^{-1} \frac{1}{\mathrm{E}[Z^{\mathrm{M}}]}. \tag{A.13}$$

Clearly,

$$\lim_{C\to\infty} \frac{1}{C\alpha \mathrm{E}[BP^{[C,f]}_{\text{ser}}]} = 0,$$

and so the limit of $\lambda_r^{[C,f]}$ depends on the rates at which $\mathrm{E}[BP^{[C,f]}_{\text{swi}}]$ and $\mathrm{E}[BP^{[C,f]}_{\text{ser}}]$ increase with $C$. If they increase at a comparable rate, i.e.,

$$\lim_{C\to\infty} \frac{\mathrm{E}[BP^{[C,f]}_{\text{swi}}]}{\mathrm{E}[BP^{[C,f]}_{\text{ser}}]} = d > 0,$$

then

$$\lambda_r^{[\infty]} = \left(\frac{1}{1+d}\right) \frac{1}{\mathrm{E}[Z^{\mathrm{M}}]} < \frac{1}{\mathrm{E}[Z^{\mathrm{M}}]},$$

implying a strict inequality in Equation (3.10) after applying Little's Law and Equation (A.6). It also follows that if

$$\lim_{C\to\infty} \frac{\mathrm{E}[BP^{[C,f]}_{\text{swi}}]}{\mathrm{E}[BP^{[C,f]}_{\text{ser}}]} = 0,$$

then Equation (3.10) is an equality.

□

## A.5 Algorithm for Section 3.5.3: Smart Bernoulli Optimization

Letting precision $\in \mathbb{Z}^+$ denote the number of decimal places we are interested in approximating to and $\mathrm{E}[N_{\mathrm{W}}](p_2^{\mathrm{SB}})$ represent the expected number of working machines as a function of $p_2^{\mathrm{SB}}$, we apply:

```
start = 0
size = 0.1
steps = 11
For $i = 1, 2, \ldots,$ precision:
    For $j = 1, 2, \ldots,$ steps:
        $p_{2,j}^{\mathrm{SB}} = \text{start} + (j-1) \times \text{size}$
        $E_j = \mathrm{E}[N_{\mathrm{W}}](p_{2,j}^{\mathrm{SB}})$
    $j_m = \{j \in \{1, 2, \ldots, \text{steps}\} : E_j = \max_k\{E_k\}\}$
    if$(p_{2,j_m}^{\mathrm{SB}} > 0)$
        start = $p_{2,j_m}^{\mathrm{SB}} - \text{size}$
        if$(p_{2,j_m}^{\mathrm{SB}} < 1)$ steps = 21
    size = size/10
$\hat{p}_2^{\mathrm{SB}} = p_{2,j_m}^{\mathrm{SB}}$
```

What this algorithm does in iteration $i \in \{1, 2, \ldots, \text{precision}\}$ is divide an interval of probabilities into increments of width $10^{-i}$, solve for $\mathrm{E}[N_{\mathrm{W}}]$ at each $p_2^{\mathrm{SB}}$ which separate the increments and determine which of these resulted in the maximum value, then restart the loop for the next $i$ investigating an interval with length $2 \times 10^{-i}$ centered around that probability, or if it is a boundary value of 0 or 1, an interval of length $10^{-i}$ including the said boundary. The above is a condensed version of the algorithm for readability and space considerations, which may have its efficiency improved slightly by being altered to not re-calculate $\mathrm{E}[N_{\mathrm{W}}]$ at any previously considered $p_2^{\mathrm{SB}}$'s. We do not propose this algorithm for its speed, but rather for its accuracy to a given decimal place without the need of derivatives, and the fact that it is able to return a probability of exactly 0 or 1.

## A.6 Derivation of Absorption Probabilities

We will consider the derivation of the absorption probabilities for absorbing Markov chains in discrete and continuous time. To begin, consider a reducible DTMC $\{X_n, n \in \mathbb{N}\}$ having transient states $0, 1, \ldots, M-1$ and absorbing states $M, M+1, \ldots, N$. We can write the TPM of such a DTMC in terms of four blocks, i.e.,

$$P = \begin{array}{c|c|c} & 0 \quad 1 \quad \cdots \quad M-1 & M \quad M+1 \quad \cdots \quad N \\ \begin{matrix} 0 \\ 1 \\ \vdots \\ M-1 \end{matrix} & Q & R \\ \hline \begin{matrix} M \\ M+1 \\ \vdots \\ N \end{matrix} & \mathbf{0} & I \end{array}.$$

Here, $Q$ contains the one-step transition probabilities between transient states, $R$ contains one-step transition probabilities from transient states into absorbing states, $\mathbf{0}$ is a matrix of zeroes, and $I$ is an identity matrix (such that the transition probability from an absorbing state back to itself equals 1).

Let $T$ denote the random time of absorption, such that

$$T = \min\{n \in \mathbb{N} : X_n \in \{M, M+1, \ldots, N\}\}.$$

Additionally, let $T_i$, $i = 0, 1, \ldots, M-1$, denote the *remaining number of transitions* until absorption given that the DTMC is in state $i$. If we let $i$ and $k$ both denote transient states, then it is clear that by definition

$$T|(X_0 = i) \sim T_i,$$

while we also have

$$T|(X_1 = k, X_0 = i) \sim 1 + T_k,$$

which follows from the Markov property and stationary assumption of DTMCs.

Now, let us define $U_{i,j}$ as the probability that the DTMC will be absorbed into state $j \in \{M, M+1, \ldots, N\}$ given that it is currently in transient state $i \in \{0, 1, \ldots, M-1\}$. Conditioning on the first transition of the DTMC,

$$U_{i,j} = P(X_T = j|X_0 = i) = \sum_{k=0}^{n} P(X_T = j|X_1 = k, X_0 = i)P(X_1 = k|X_0 = i). \tag{A.14}$$

Note that for $k \in \{M, M+1, \ldots, N\}$,

$$P(X_T = j|X_1 = k, X_0 = i) = \delta_{k,j},$$

while for $k \in \{0, 1, \ldots, M\}$,

$$\begin{aligned} P(X_T = j|X_1 = k, X_0 = i) &= P(X_{1+T_k} = j|X_1 = k, X_0 = i) \\ &= P(X_{1+T_k} = j|X_1 = k) \\ &= P(X_{T_k} = j|X_0 = k) \\ &= P(X_T = j|X_0 = k) = U_{k,j}. \end{aligned}$$

Thus, Equation (A.14) simplifies to

$$U_{i,j} = \sum_{k=0}^{M-1} P_{i,k} U_{k,j} + P_{i,j} = \sum_{k=0}^{M-1} Q_{i,k} U_{k,j} + R_{i,j}, \tag{A.15}$$

which in matrix form is equivalent to

$$U = QU + R,$$

implying that

$$U = (I - Q)^{-1} R, \tag{A.16}$$

where $U$ is the $M \times (N - M + 1)$ matrix whose $(i, j)^{\text{th}}$ element is $U_{i,j}$.

Now suppose that we have a CTMC $\{X(t), t \geq 0\}$ with the same breakdown of transient and absorbing states, having infinitesimal generator matrix

$$Q = \begin{array}{c|c|c} & 0 \quad 1 \quad \cdots \quad M-1 & M \quad M+1 \quad \cdots \quad N \\ \begin{matrix} 0 \\ 1 \\ \vdots \\ M-1 \end{matrix} & S & S_0 \\ \hline \begin{matrix} M \\ M+1 \\ \vdots \\ N \end{matrix} & \mathbf{0} & \mathbf{0} \end{array}.$$

The key to this analysis is that a CTMC will be absorbed into the same state as its embedded DTMC. Converting the infinitesimal generator matrix of a CTMC into the TPM of its embedded DTMC is easier if $v_i = v$ for all $i \in \mathcal{S}$. However, in general, a CTMC will have varying sojourn rates $v_i$ in each transient state.

To account for this, we may apply uniformization (e.g., Ross [82], Section 6.8) to modify the stochastic process by including the possibility to transition from a state back to itself. This is achieved by the addition of an extra exponential timer to each state $i$ with rate $v - v_i$, $v \geq \max\{v_i, i \in \mathcal{S}\}$, bringing the total exponential rate for the time between (potential) transitions to $v$ for every state. The TPM for the modified process's embedded DTMC is

$$P^* = I + \frac{1}{v} Q = \left[ \begin{array}{cc} I + \frac{1}{v} S & \frac{1}{v} S_0 \\ \mathbf{0} & I \end{array} \right],$$

which is itself an absorbing DTMC. Applying Equation (A.16), the matrix of absorption probabilities is

$$U^* = \left( I - \left( I + \frac{1}{v} S \right) \right)^{-1} \frac{1}{v} S_0 = -S^{-1} S_0. \tag{A.17}$$

Relative to the embedded DTMC of $\{X(t), t \geq 0\}$, we have simply added the possibility of transitioning from transient states to themselves. Since the process changes state in a given transition, the probability that it transitions from $i$ to $j$ can easily be shown to equal $P_{i,j}$ from the original embedded DTMCs TPM. Therefore, while it may require more transitions to reach an absorbing state, its absorption probabilities will be identical. Thus, Equation (A.17) must also represent the matrix of absorption probabilities for $\{X(t), t \geq 0\}$.