Layered Queueing Networks

Performance Modelling, Analysis and Optimisation

A catalogue record is available from the Eindhoven University of Technology Library.
ISBN: 978-90-386-3763-1

Printed by Ipskamp Drukkers, Enschede, the Netherlands.
Cover design by Bas Ruhé.

Layered Queueing Networks

Performance Modelling, Analysis and Optimisation

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de
Technische Universiteit Eindhoven, op gezag van
de rector magnificus prof.dr.ir. C.J. van Duijn,
voor een commissie aangewezen door het College
voor Promoties, in het openbaar te verdedigen
op dinsdag 17 februari 2015 om 14:00 uur

door

Jan-Pieter Lodewijk Dorsman

geboren te Amstelveen

Dit proefschrift is goedgekeurd door de promotoren en de samenstelling van de promotiecommissie is als volgt:

voorzitter:	prof.dr. E.H.L. Aarts
1e promotor:	prof.dr.ir. O.J. Boxma
2e promotor:	prof.dr. R.D. van der Mei (VUA en CWI)
copromotor:	dr. M. Vlasiou
leden:	prof.dr. R.J. Boucherie (UT)
	prof.dr.ir. R. Dekker (EUR)
	prof.dr. A.G. de Kok
	prof.dr.ir. J. Walraevens (Universiteit Gent)

DANKWOORD (ACKNOWLEDGEMENTS)

Dit proefschrift is het resultaat van enkele jaren onderzoek verricht als promovendus aan de Technische Universiteit Eindhoven (TU/e) en het Centrum Wiskunde & Informatica (CWI) te Amsterdam. Tijdens deze periode heb ik mij gesteund geweten door velen, die op een directe dan wel indirecte wijze een essentiële bijdrage hebben geleverd aan de totstandkoming van deze dissertatie. Het is niet meer dan gepast aan hen in dit welgemeende dankwoord mijn dank te betuigen.

Allereerst ben ik zowel mijn copromotor Maria Vlasiou als mijn promotoren Onno Boxma en Rob van der Mei zeer erkentelijk voor de uitmuntende begeleiding, die zij mij op complementaire wijze geboden hebben. Maria, jouw kritische blik heeft mij dikwijls verder doen kijken dan ik in eerste instantie deed. Ik heb veel van jou geleerd en die wijze lessen zullen ongetwijfeld in de toekomst nog vele malen hun dienst bewijzen. Onno, ik heb een voorbeeld mogen nemen aan jouw manier van en jouw visie op het verrichten van onderzoek, en wat dat betreft beschouw ik mijzelf als bevoorrecht. Ook de uitzonderlijk intensieve maar niettemin buitengewoon prettige samenwerking bij vele onderwijstaken zal mij nog lang heugen. Rob, jouw tomeloze enthousiasme heeft ervoor gezorgd dat ik het plezier in mijn werkzaamheden heb behouden. Ik heb veelvuldig van jouw ijzersterke relativeringsvermogen kunnen profiteren, hetgeen bij tijd en wijle van groot belang is geweest.

Naast de hierboven genoemde personen heb ik in mijn promotietijd onderzoek mogen verrichten met vele anderen. In het bijzonder ben ik René Bekker, Sandjai Bhulai, Sem Borst, Nir Perel, Petra Vis, Erik Winands en Bert Zwart dankbaar voor ettelijke prettige samenwerkingsverbanden. Deze hebben geleid tot verscheidene onderzoeksresultaten, waarvan enkele in dit proefschrift zijn opgenomen. Voorts ben ik dank verschuldigd aan Richard Boucherie, Rommert Dekker, Ton de Kok en Joris Walraevens voor het zitting nemen in de promotiecommissie en de grondige beoordeling van het manuscript. Bij het schrijven van dit proefschrift heb ik veel technische hulp genoten van Marko Boon en de omslag van deze dissertatie is ontworpen door Bas Ruhé. De Nederlandse Organisatie voor Wetenschappelijk Onderzoek (NWO) ben ik erkentelijk voor de financiële ondersteuning in het kader van een project van het stochastiekcluster STAR.

Om op succesvolle wijze werk te verzetten in het wetenschappelijke metier is een prettige werkomgeving vereist. Mijn collega's op het CWI, inclusief de masterstudenten die in de voorbije jaren bij het CWI stage hebben gelopen, hebben in niet geringe mate ervoor gezorgd dat aan deze voorwaarde is voldaan. Het spijt mij oprecht dat ik niet ieder van hen apart bij naam kan noemen, daar dit een oeverloze stroom van namen zou genereren. Wel kan ik zonder enige twijfel stellen dat de collegiale sfeer in de stochastiekgroep onontbeerlijk is gebleken bij de voltooiing van dit proefschrift. De reuring en

de vele karakteristieke curiositeiten die aan de orde van een doorsnee werkdag op het CWI zijn geweest, hebben de onderzoeks- en onderwijsbeslommeringen aanzienlijk veraangenaamd. In het licht van een prettige werkomgeving dien ik ook de collega's van EURANDOM en de stochastieksectie bij de TU/e, inclusief het secretariaat, te noemen. Ongeacht de duur van elke periode waarin ik verzuimde de rivieren over te steken, heb ik mij bij elk daaropvolgend rentree in Eindhoven weer welkom en thuis gevoeld. Dit is niet zonder meer vanzelfsprekend en ik ben hun hiervoor dankbaar. Ik ben de afgelopen jaren op deze wijze onderdeel geweest van twee relatief grote onderzoeksgroepen, elk met zijn eigen populatie en karakteristieken. Ik beschouw dit als een verrijking.

In het laatste jaar van mijn promotietijd heb ik enkele maanden de onderzoeksgroep Stochastic Modelling and Analysis of Communication Systems (SMACS) mogen bezoeken, die onderdeel is van de vakgroep Telecommunicatie en Informatieverwerking (TELIN) van de Universiteit Gent in België. Dit verblijf was uiterst leerzaam in meerdere opzichten en bovenal zeer naar mijn goesting. Ik dank Dieter Claeys en Joris Walraevens voor de hoffelijke Vlaamse gastvrijheid en de mij geboden mogelijkheid mee te draaien in het Gentse onderzoeksleven. Naast hen heb ik tijdens mijn verblijf in West-Vlaanderen ook op een zeer prettige wijze onderzoek verricht met Dieter Fiems, Wouter Rogiest en Jasper Vanlerberghe.

Ten laatste, maar beslist niet ten minste, wens ik een woord van dank te richten aan familie en vrienden voor het verlenen van het constante besef dat er meer bestaat dan onderzoek. Voor een directe bijdrage aan de totstandkoming van dit proefschrift kan en zal ik hen dan ook zeer zeker niet bedanken. Ondanks dit, of liever gezegd dankzij dit, is de succesvolle completering van deze dissertatie wellicht nog het meest aan hen toe te schrijven.

Jan-Pieter Dorsman
Amsterdam, november 2014

CONTENTS

1

INTRODUCTION

1.1 Motivation

In today's society, queueing phenomena arise in many situations. In service facilities, it frequently occurs that requested services cannot be provided immediately to users. This results, for example, in waiting lines at counters, elevators and traffic lights, or queues in a less physical sense in call centers or in healthcare. Less obvious examples of such congestion phenomena arise in communication systems and computer networks, where data has to be transported from one place to another. One can think of the internet, where continuous capacity improvements have to be made in order to keep up with the fast growing demand. Although delays in transportation of information in this context occur on a completely different time scale (e.g. in the order of milliseconds), they are not any less serious for the users. The existence of all these undesirable congestion effects has led to the development of a mathematical discipline that studies queueing phenomena so as to answer optimisation questions such as how to allocate resources in order to avoid queueing as much as possible. This thesis is placed in the context of this discipline, which goes by the name of *queueing theory*.

At an abstract level, the queueing models that arise from the study of congestion phenomena consist of *queues*, at which *customers* arrive. The customers wait in the queue until they can receive the service that they require from a *server*. Queueing models are typically of a stochastic nature. Namely, the durations of the interarrival times and service times of the successive customers are not exactly specified, but they are assumed to have some (known) probability distribution. This reflects the fact that in most applications, it is uncertain when demand arises for any type of service or how large that demand will be.

Since the first paper on queueing theory in 1909, in which A. K. Erlang performed a systematic study of the dimensioning of telephone switches [94], there has been a vast body of literature that is concerned with a wide array of queueing models. Perhaps the best-studied and most elementary queueing model consists of customers arriving at a single queue that is served by a single server at a constant service speed. This single-server queue has been studied very extensively (see e.g. [67]). The results obtained for this model so far have contributed to a better understanding of queueing systems in general. The analysis of queueing models may, however, be challenging. This is illustrated by

the fact that under general conditions, the waiting time of a customer in a single-queue single-server model is not understood completely. On top of that, in many applications, even the assumptions of a single queue or a single server are not valid.

An example of such an application can be found in the area of manufacturing systems. In the process from raw material to an end product, a part in a manufacturing plant may be stored in intermediate buffers multiple times, waiting for the next phase of processing. Each of these buffers contains parts that need to be processed by a specific work station, which is typically specialised in providing one such phase of the production process. In manufacturing systems, one is often interested in the total time it takes for raw materials to be converted into end products. Such performance measures may be analysed by modelling the production process as a *queueing network*. Queueing networks are models that typically consist of multiple queues, each of which are served by a number of servers. After spending time in a queue and undergoing a subsequent phase of service, a customer does not necessarily leave the system, but may be routed to any other queue in the network, or even rerouted to the same queue, in order to receive yet another phase of service. The first result on queueing networks was presented by [130] in 1957. This work spurred a significant interest in these networks, leading to many other seminal results, such as those found in [30, 106, 137, 156, 221]. For an overview of the literature on queueing networks, see e.g. [48, 60, 138].

This dissertation is concerned with the mathematical study of a particular type of queueing networks. Recent applications in engineering, business and the public sector led to systems with complex, often layered, service architectures, where there exist service providers that, at times, may require service themselves from other servers. An appealing example of this architecture is formed by peer-to-peer networks, where users do not only provide service to other users by uploading their files, but also request service by downloading files from other users. One may also again think of a manufacturing setting, where machines processing products in parallel are served by an operator. Yet another example is given by large call centers, of which the organisational structure is often multi-layered in the sense that an agent handling external calls may have to consult a more senior agent in case of a complicated query (cf. [148]). We also mention the application of container transshipment in terminals, where (possibly automated) vehicles transport containers from a ship to a terminal or vice versa. Upon reaching either destination, a vehicle needs to wait for a crane to unload the container it carries or to load a new container onto it in order to resume service (cf. [39, 79, 86]). Other areas where similar layered architectures exist include healthcare [266] and, as we will see in the sequel, many applications in computer science.

The need for the analysis of these important layered structures leads to the formulation of *layered queueing networks*. These queueing networks consist of multiple layers, where the servers of any layer act as customers of the layer directly below. Thus, in layered queueing networks, the entities do not necessarily act strictly as customers or servers, but they may assume both roles simultaneously or consecutively. Mathematical analysis of these networks is challenging, since the interaction between the layers may be significant and thus must be taken into account. For example, the performance of lower-layer servers may heavily affect the congestion levels incurred by higher-layer customers.

In this thesis, we perform an in-depth study of three such layered queueing networks, where the interactions between the layers cannot be ignored. We develop several methods for the performance analysis and optimisation of each of these models, which take

their interactions into account. With these methods, we aim to gain insight into the impact of the layer interactions on the performance and control of the queueing networks considered.

The remainder of this chapter is organised as follows. In Section 1.2, we give an account of the current body of literature on the performance modelling and analysis of layered queueing networks. Section 1.3 subsequently presents a detailed model description of the three layered queueing networks that we consider in this dissertation. Finally, we outline the contributions of the thesis in Section 1.4, and we give an overview of how the subsequent chapters of this thesis are structured.

1.2 Literature review of layered queueing networks

Despite the ubiquity of queueing phenomena with a layered structure in many disciplines, previous studies of layered queueing networks are almost exclusively restricted to the computer science literature, e.g. for the study of decentralised systems with nested resource possession or peer-to-peer networks. In these studies, several approximate and heuristic methods are derived to analyse the dynamics of these applications. The in-depth mathematical analysis of layered queueing models in general is, however, almost completely an uncultivated area of research, except for a few initial studies that analyse models which are very rudimentary when compared to the layered queueing phenomena occurring in practice. Due to the layered character, however, a detailed analysis of these stylised models already turns out to be very challenging. In Section 1.2.1, we give a brief overview of the work performed in the context of computer systems and software engineering, where immensely complex layered structures are analysed by means of heuristic methods. Subsequently, Section 1.2.2 discusses the initial queueing-theoretical studies of stylised models, which are not necessarily tailored to solving strictly computer science related problems.

1.2.1 Computer science literature

We now give an overview of the computer science literature on layered queueing networks. The list of references presented is by no means exhaustive; it rather serves the purpose of indicating the continuing interest in the modelling of layered queueing networks in computer science. In the following, we do not present the references in a chronological order, but we group them together in several categories. This taxonomy allows for a better overview of the variety of the examined subjects.

1.2.1.1 Development of the framework of layered queueing networks and their analysis

Many studies in the context of computer networks and software engineering concern themselves with systems that have a layered architecture. For example, in the design of computer networks, an important question is how functionalities should be allocated to different layers so as to meet the criteria set by the users in terms of efficiency, robustness and other matters [61]. A possible answer to this question is given by the Open Systems Interconnection (OSI) model [235, Section 1.4.1], but many such allocations

are possible. To aid in this kind of decision making, an extensive body of literature exists that centers around the performance modelling and analysis of decentralised systems with nested resource possession. These systems are modelled in the framework of layered queueing networks, where entities that provide service in one layer can request service from servers in lower layers.

In the computer science literature, layered queueing networks were first introduced as *active-server models* in [278, 280], which include the key property that a server may pause during its service for a nested request to another server in a lower layer. These models have been extended in [279] to *stochastic rendezvous networks* that allow for different types of service, where also an approximate solution method for this type of networks is obtained based on the Bard-Schweitzer approximate version [29, 219] of the mean value analysis algorithm [207] known from theory on regular queueing networks. A similar network is studied in [214], where the *method of layers* is proposed to analyse the performance of the layered system. This method is based on a tailored version of another approximate algorithm based on mean value analysis, namely the lineariser algorithm [58]. The method of layers is a development from the *lazy-boss algorithm* [211], of which the name is based on the comparison of a server waiting for a lower-layer service with a boss waiting for his employee to finish his own work before continuing to do something else.

Since this initial flow of papers, many extensions to the framework of layered queueing networks have been considered. We mention the extension of deferred service, where a lower-layer server may have to complete a second part of service after ending the service from a customer's point of view [101], and that of fair-share queueing, where an effort is made to incorporate customer fairness into the layered queueing framework [160]. Incorporation of performance degradation into the framework as a result of 'aging' of software and hardware components is considered in [75]. Another extension that has been studied is that of quorum patterns, where a server, after sending out N requests to a lower layer, already proceeds operating after $J < N$ of these requests are completed [15]. Finally, inclusion of management components in the model for the automatic detection of software and hardware failures and subsequent reconfiguration has been a well-studied topic [73, 74, 76, 77]. In an effort to incorporate most of the work mentioned above in a single model, [100] attempts to unify many model variations and extensions into one general framework, and presents a general solution technique adapted to it, which is again based on mean value analysis.

Apart from the modelling point of view, much attention has also been paid to the refinement of the initial performance prediction methods found in the seminal work of [214, 279]. For example, in [16, 17], it is explained how the exploitation of any symmetric properties in the system can lead to an increase in computational efficiency. The enrichment of techniques based on mean value analysis with (non-)linear programming methods for performance prediction purposes has been considered in [159, 165, 166]. Several other computational techniques, which are not based on mean value analysis, are covered in [204], where an alternative approximation algorithm is derived by drawing a parallel with queueing networks with blocking (see e.g. [28]), and in [122, 241, 242], where stochastic process algebras are used to analyse the performance of the system. Furthermore, the so-called weighted-average method derived in [151] uses simulation techniques for performance prediction purposes. The computation of tail probabilities of the complete distribution of the response times rather than just their means has been

studied in [291, 292]. Other performance prediction methods have been developed in [142, 152, 176]. Although perhaps diverse in character, all of these computational methods have in common that they are only approximate and heuristic in nature. Finally, we mention the studies [24, 25, 26, 27], where the performance prediction as a result of modelling systems as a layered queueing network is compared to prediction methods based on the analysis of historical data.

1.2.1.2 Development of a layered queueing model

In the design of a software system, it is commonly believed that performance analysis should be integrated into the development process as early as possible as opposed to the more frequent so-called *fix-it-later approach* that postpones performance concerns until the system is completely implemented [226]. The reason is that a failure to detect performance pitfalls in a system in its earliest state of development may turn out very costly. Therefore, a significant amount of attention has been paid to the problem of how to build performance models based on the layered queueing network formalism from a description of the system's architecture. For example, [197] proposes a formal approach for this translation using graph transformations. An automated approach towards the construction of performance models of software systems based on the traces of behaviour of the systems, prototypes or executable models is proposed in [128].

In the design of a software system, the first definition of the system may be given in a Use Case Maps (UCM) notation (see e.g. [57]). Particular attention has thus been given to the important problem of transforming UCM scenario models into layered queueing models [192]. The steps needed to achieve such a transformation are given in [191], and [193, 199] describe tools for the automation of this process. Another important standard in the specification of software systems, including their structure and design, is the Unified Modelling Language (UML) [216]. The transformation from UML specifications to layered queueing networks is a topic first studied in [134], and since then, it has been considered extensively (see e.g. [71, 198, 284]). Several approaches for the translation from UML have been proposed based on a graph grammar-based method [112, 195] and the so-called XSLT language [89, 111]. Transformation approaches specifically tailored to software product line models, models with non-functional security aspects and aspect-oriented models are studied in [236], [200, 281] and [196], respectively.

Apart from UCM and UML, model transformations from Palladio component models and the so-called Specification and Description Language have been studied in [150] and [92], respectively. In [90, 91], an attempt is made to combine different standards and proposed frameworks into one automated unified tool.

Following the problem of how to construct a layered queueing network from a design, the issue arises how to adapt the design and search the design space for the best performance possible without excessive computational efforts. This issue is discussed in [174, 178, 210].

1.2.1.3 Estimating model parameters

Next to the modelling of a design as a layered queueing network and the subsequent tuning, another important problem, which needs to be addressed to secure a successful performance evaluation of the system at any point of the software development process, constitutes the correct measurement of model parameters, such as the resource demands

of the servers in higher layers. In [215], two methods are discussed for the estimation of the resource demands in application services. In the first method, resource consumption is measured directly for each service request to each lower-layer service. The second methods entails the performance measurement for the service process as a whole (including multiple service requests to lower layers) and the use of statistical techniques afterwards to estimate the resource demands of the services individually. Both methods are compared and appear to show a clear trade-off between accuracy and feasibility. After this early study, many regression-based methods for predicting resource demand in multi-layered systems have been discussed, such as those in [213] and in [290]. Alternatively, it is shown in [282, 283, 293] how Kalman filtering can be used to track changes in the parameters of the layered queueing model. Finally, we mention [186], where an online method for the dynamic estimation of the resource demands is devised that is specifically tailored to implementation in web servers, which tend to be of a very large scale.

1.2.1.4 Applications

A large number of studies has been devoted to the modelling of a specific application as a layered queueing network. For example, [83, 258] successfully analyse the client response times in web servers and derive several sensitivity properties (e.g. with respect to the number of available servers or the network latency). Another example is that of middleware, which is often used in distributed systems to provide interoperability between the various components of the system. In [168, 194, 262], middleware systems are analysed at different levels of abstraction. Furthermore, there are several studies that concern themselves with the advantages and the shortcomings of the layered queueing framework for the development of enterprise resource planning software [107, 212, 237] and enterprise application software in general [169, 240, 246, 285, 286]. Similarly, [72, 243] discuss the modelling of service-oriented architectures and enterprise resource planning software, respectively, as layered queueing networks. Finally, to underline the ubiquity of systems with a layered structure in computer science, we mention that the framework of layered queueing networks has been applied for the performance evaluation and optimisation of database management systems [203], e-commerce applications [164], physically mobile systems with highly dynamic user mobility [81], telecommunication software systems [224] and virtual machine technology [133].

1.2.1.5 Miscellaneous

So far, we have given an overview of the studies performed in the computer science literature in the framework of layered queueing networks, and their application to systems with a clear layered structure. However, the performance analysis of layered queueing networks also has less apparent applications than those mentioned previously. As an example, we mention peer-to-peer networks. These decentralised networks are used for file sharing between users and are based on the principle that users downloading a file from the network themselves contribute their upload bandwidth to allow others to download pieces of the file they already downloaded.

Although peer-to-peer networks clearly consist of entities that act as both *customer* and *server* by downloading and uploading files concurrently, they violate the assumption that resource possession in the network is nested. In other words, there is no way to divide these networks in layers such that a server from one layer only requests service

from a layer directly below. Despite this violation, however, [222] shows that peer-to-peer networks can still be modelled in the layered paradigm. Peer-to-peer networks have spurred a lot of interest, resulting in a separate body of literature concentrating on the performance analysis of peer-to-peer networks. We refer to [114, 294] and references therein for an overview of this literature and for a study of the stability of such a system. In the literature on peer-to-peer networks, many complicated questions are addressed, such as how the files should be divided into pieces so as to expedite the dissemination of files [217] and how the dynamics of the system change when a mechanism is implemented that allocates download bandwidth to users proportional to their upload speed [187]. However, also here, an exact analysis of the download times for the users is still lacking.

1.2.2 Queueing literature

As mentioned before, there hardly exist any in-depth queueing-theoretical studies where servers of one layer can act as customers in another layer. This is perhaps because even the simplest layered models do not always allow for simple solutions. Interactions and dependencies between layers, even when they are a consequence of seemingly simple model features, often complicate the analysis considerably, possibly up to a point where an exact analysis is out of reach. We mention a few exceptions found in the literature, where simplified layered models are analysed in detail. Even though the level of simplification with respect to the layered architectures found in practice is significant, this does not directly imply that the analysis in these studies is trivial.

An example of a 'simple' model is the following two-layered model where the first layer consists of a single queue with N servers. The servers of this queue act as customers in the second layer in the sense that, in turn, these servers receive service resources from a single second-layer server in a processor-sharing fashion. This model is equivalent to the so-called limited processor-sharing queue, i.e. a queue of which the first N customers are each served concurrently at a service rate of $\frac{1}{\min\{k,N\}}$ when there are k customers in the system and the remaining $\max\{N-k,0\}$ customers receive no service at all. Even for this model, an exact analysis is far from trivial, judging from the fact that the literature on the limited processor-sharing queue focuses on approximations [22, 287, 288, 289], stochastic ordering results [183] and asymptotic results [179].

Several generalisations of this layered model have been studied. An example of this is the case where the first layer does not consist of a single multi-server queue, but two multi-server queues in a tandem configuration. The servers of both queues still act as customers in the second layer, where they receive service in a processor-sharing fashion. In [254], the stability and the throughput of this extended model is studied, whereas [255] investigates the static optimisation problem of how to divide the first-layer servers over the queues so as to minimise the expected sojourn time of first-layer customers. For the case of two first-layer queues in tandem or in parallel, necessary and sufficient conditions for a product-form solution to exist are derived in [256].

This model has been generalised further to allow for an arbitrary number of first-layer queues. For a model in which the first-layer queues are placed in tandem, [250] considers the problem of how to assign the first-layer servers statically to the first-layer queues so as to maximise the throughput of the system. A similar, but dynamic assignment problem with the goal of minimising the expected sojourn time is studied in [253]. When we drop any assumption on the configuration of the first-layer queues, so that the queues do not

necessarily have to be placed in tandem or in parallel, stability results on the resulting model can be found in [132]. This paper not only considers the case where the first-layer servers (i.e. the second-layer customers) are assigned an equal rate of service from the server in the second layer, but these service rates may differ mutually depending on the numbers of first-layer customers waiting in each individual first-layer queue. When restricting to processor-sharing service in the second layer, it is shown in [268] that a separation of time scales occurs in heavy traffic. That is, the first layer and the second layer work on different time scales when the system is under critical load and each layer views operations at the other layer as if they were constant.

The final layered network encountered in the queueing literature that we discuss again seems strikingly simple, but is in fact analytically far from trivial to analyse. As before, this network consists of two layers. Each of these layers is comprised of an $M/M/\cdot/\cdot$ type queue. The distinguishing feature of this model is that the customers present in the first queue act as servers of the second queue. In [189], probability generating functions for the steady-state queue length distributions are derived for two variants of this network. In both variants, the first layer entails a regular $M/M/1$ queue. The second layer of the first variant also constitutes a single-server queue. Its service rate is, however, not constant, but scales linearly with the number of customers present in the queue of the first layer (i.e. the first-layer customers work together on serving one second-layer customer). In the second variant of the model, the queue of the second layer is an $M/M/N$ type queue where the number of servers is not constant, but in fact equals the varying number of customers in the first layer (i.e. the first-layer customers each serve a different second-layer customer). In a follow-up project, the authors of this work even increase the complexity of their model by adding the feature that the customers of the second queue now also act as servers of the first queue (cf. [190]). This creates an interaction between the layers in two directions, which complicates the analysis even further. As a result, the authors resort to the usage of matrix-analytic methods (see e.g. [180]) to compute performance measures such as the mean queue lengths.

1.3 Model descriptions

This dissertation consists of three parts, each of which provides a detailed study of a particular layered queueing network. We refer to these layered queue networks as the *extended machine repair model*, the *Markovian polling model* and the *carousel storage model*. In this section, we provide a detailed description for each of these models.

1.3.1 The extended machine repair model

The first layered queueing network that we consider in this thesis constitutes an extension of the classical machine repair model. This model, also known as the computer-terminal model (cf. [33]), the time-sharing system (cf. [143, Section 4.11]) or the machine-interference problem, is well studied in the literature. In the machine repair model, there is a number of machines working in parallel and one repairman. The machines are working independently, and as soon as a machine fails, it joins a repair queue in order to be repaired by the repairman. It is one of the key models to describe problems with a finite input population. A fairly extensive analysis of the machine repair model can be found

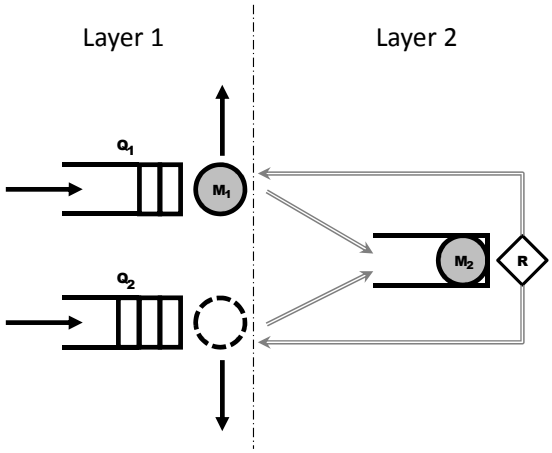

FIGURE 1.1: The extended machine repair model.

in Takács [232, Chapter 5], and surveys reviewing the extensive literature on this model can be found in [116, 228].

So far, the effects of the repairman's performance on the machine's availability have been studied extensively, but the question of how the repairman's performance affects the backlogs of products to be processed by the machines has hardly been considered. Motivated by this, we study the queues of products waiting to be processed by the machines. This naturally leads to the formulation of a two-layered queueing network, as the machines now have the dual role of both a customer and a server. As in the traditional model, the machines have a *customer role* with respect to the repairman, but they now also have a *server role* with respect to the products.

The first layer of the resulting layered queueing network contains the queues of products; see Figure 1.1. Each of these queues is served by its own machine. At any point in time, a machine is subject to breakdowns irrespective of the state of the first-layer queue. When a machine breaks down, the service of a product in progress is interrupted and either restarted or resumed once the machine becomes operational again. For ease of discussion, we often assume that, as opposed to the classical machine repair model, there are two machines only. As will become evident, the methods that we apply are readily extended to more machines or repairmen, but certain computations become increasingly cumbersome.

The second layer consists of a repairman and a repair buffer. If, upon breakdown of a machine, the repairman is idle, the machine is immediately taken into service. Once the machine is again operational after the necessary repair time, it starts serving products once more. However, when the repairman is busy repairing another machine, the machine waits in the buffer. In that case, only when the repair of the other machine has been completed, the repair of the current machine starts. The downtime of the machine then not only consists of the necessary downtime, but also of a waiting time.

This extension of the machine repair model has immediate applications in manufacturing. Therefore, we will throughout refer to the entities in the model as *products*, *machines* and the *repairman*, respectively. The extended model is, however, also of interest in other

application areas, such as telecommunication systems. For instance, the extended machine repair model occurs naturally in the modelling of middleware technology, where multi-threaded application servers compete for access to shared object code. Access of threads to the object code is typically handled by a portable object adapter that serialises the threads trying to access a shared object. During the complete duration of the serialisation and execution time, a thread typically remains blocked, and upon finalising the execution, the thread is deactivated and ready to process pending requests [117]. In this setting, the application servers and the shared object code are analogous to the machines and the repairman, respectively, in the machine repair model.

As mentioned above, virtually all studies so far on the machine repair model have only considered the second layer as depicted in Figure 1.1 in isolation. The only exception is [269], where the queue length distributions of a queue in the first layer is approximated by drawing a connection with a single-server vacation queue. The vacation times are assumed to have their first two moments equal to those of the downtimes of the machines, but more importantly, the vacation times are assumed to be mutually independent. In the context of the extended machine repair model, it is thus assumed that there are no interactions between the two layers or even that there are no correlations between the queue lengths in the first layer itself.

An important feature of this model is the fact that machines compete for repair facilities. This introduces interaction between the two layers, with significant positive dependencies in the downtimes of the machines as a result. If the downtime of one machine is very large, its repair is probably taking longer than usual, increasing the likelihood for the other machine to break down in the meantime. As a result, the queue lengths of the first-layer queues exhibit correlations of an unknown form. In fact, consecutive downtimes of a machine in isolation are also correlated. Because of the increased likelihood of the other machine to break down, the next downtime of the one machine is probably larger too. These correlations cannot be disregarded, as it turns out that they have a considerable impact on the waiting times. Therefore, the interactions between the two layers cannot be ignored and the approximation derived in [269] can be improved significantly. The correlations, however, are not well understood, since they are only implicitly defined through the uptimes and the repair times of the machines.

In our analysis, we explicitly take the interaction between the layers and the correlations between the first-layer queues into account. However, the dependence between these queues makes exact analysis of the queue length distributions difficult. The amount of work present in a first-layer queue can in principle be modelled as a reflected Markov additive process (see [19, Section XI.2] for a definition), but its distribution is not easily derived from that. Numerical evaluation, e.g. by simulation, may also be challenging. Especially when the model involves breakdowns and repairs that occur on a larger time scale than actual product arrivals and services, the computation time needed to achieve accurate results may be unacceptably long. Additional difficulties arise since we allow the machines to have mutually different uptime and repair-time distributions. For example, as observed in [109], the arrival theorem (cf. [156]) cannot be used anymore to derive the stationary downtime distributions of the machines. Despite these technical complications, we derive several powerful approximations for the expectations and the complete distributions of the queue lengths of the first-layer queues. We also study the question of how to allocate the repairman's resources optimally so as to minimise these queue lengths as much as possible.

1.3.2 The Markovian polling model

In the second part of this thesis, we study a queueing network consisting of multiple queues attended by a single server as depicted in Figure 1.2. The server visits the queues in some order to render service to the customers waiting at each of the queues and incurs stochastic switch-over times when he moves from one queue to another. The order in which the server visits the queues is governed by a discrete-time Markov chain. As we will explain below, the discontinuous nature of the availability of a server at a queue introduces a layered structure in the queueing network.

This type of queueing system is commonly called a *polling system*. The first studies on polling systems originate from the late 1950s, when the papers of Mack et al. [171, 172] concerning a patrolling repairman model appeared. In a broader perspective, polling models are applicable in situations where several types of users compete for access to a common resource which is available to only one type of user at a time. As such, they find their origin in many real-life applications, such as manufacturing environments and traffic systems. The polling model gained most of its popularity during the 1980s, when it turned out to be a suitable model for many computer-communication applications and protocols. For an extensive overview of the literature on polling systems and an overview of their applications, we refer to surveys such as [43, 158, 233, 263].

Many studies in the polling literature assume that the server visits the queues in a fixed, cyclic order. However, this might not be a realistic assumption in cases where the queue to be visited next is determined by an external random environment. Therefore, as stated above, we are mainly concerned with so-called *Markovian* polling systems, where the server visits the queues in an order that is governed by a discrete-time Markov chain. Thus, the order in which the server visits the queues is not necessarily a fixed order. Also, when concluding a visit period at a certain queue, it is now possible for the server to resume service at the same queue after a necessary switch-over period.

It is remarkable that in the wide body of literature, polling systems with Markovian routing have received much less attention than polling systems with conventional cyclic routing. The explanation perhaps is that the analysis of Markovian polling systems is generally considered to be of a much more complex nature than that of cyclic polling models. More specifically, it is shown in [208] that there is a striking dichotomy in the complexity of the analysis of polling systems. Polling systems of which the joint queue length process observed at time points where the server starts a visit (also referred to as polling epochs) constitutes a multi-type branching process with immigration (see e.g. [21] for a definition) are more tractable than polling systems which do not satisfy this so-called *branching property*. Due to the stochastic nature of the server routing, Markovian polling systems generally do not satisfy this branching property. Publications that deal with Markovian polling systems include [54], in which an expression for the expected amount of work in the system at an arbitrary moment is derived for a few service disciplines. This work is extended in [271], where it is shown how to derive expressions for the moments of the (joint) queue lengths for the same service disciplines. Markovian polling systems have also been studied in conjunction with theory on large deviations [80, 97] and the functional computation method [123, 124]. Furthermore, stochastic decomposition results for the queue lengths in a general class of polling systems, which covers systems with a Markovian routing mechanism, are derived in [34]. Quite a few other generalisations of the Markovian polling system have been studied in a variety of directions. For example, gated Markovian polling systems with 'semi-linear' feedback are considered in [98], [63]

discusses Markovian polling systems in which customers are blocked whenever there is already a customer in the queue and systems with retrial customers have been studied in [155]. Results for a slightly more general form of Markovian routing, where the routing probabilities may depend on the event whether a queue is empty or not, are derived in [95, 227]. Observe that the Markovian routing mechanism is very general and covers many variations of polling models studied in the literature. For instance, the cyclic polling model falls in this framework. Another example is the random routing discipline, where after any visit period, the server visits queue j with probability p_j irrespective of the queue the server just visited (cf. [144]).

The generalised way of server routing finds many applications. For instance, polling models with a Markovian routing mechanism occur naturally in the modelling of cellular data services. These services implement so-called opportunistic scheduling to profit from multi-user diversity [125, 261], which is aimed to utilise fading and shadowing of cellular users within a single cell in order to optimise bandwidth efficiency [110]. The basic idea of opportunistic scheduling is that a time slot (representing the right for transmission) is assigned to the user with the highest instantaneous signal-to-noise ratio among all users in a cell. In this way, access to the medium is randomly assigned to the multitude of users in a cell.

Another example that we will pay specific attention to can be found in the context of wireless random-access networks. So-called carrier-sense multiple-access collision-avoidance algorithms provide a common mechanism for governing the use of such a shared wireless medium in a distributed fashion. In these algorithms, the various transmitters obey random back-off times between activity periods, during which they sense the medium to avoid collisions and provide other nodes an opportunity to activate. In the case of exponentially distributed back-off durations, the alternating use of the medium by the nodes is probabilistically equivalent to Markovian routing in a polling system (or in particular, random routing). The queues and their customers in the polling model represent the packet buffers of the nodes and the packets waiting to be transmitted, respectively. Furthermore, the event of the server visiting a certain queue is tantamount to the event of the corresponding node being active. The relative values of the back-off rates induce relative priorities among the nodes, and hence a crucial question is how the back-off rates should be selected in order to minimise the overall average packet delay, which corresponds to the optimal selection of the routing probabilities in the polling system.

In addition to many other applications that can be found in the field of computer-communication systems (see e.g. [144]), Markovian polling systems may also be particularly useful in the modelling of production systems with machines processing multiple product types. The type of product that a machine should prioritise for processing at a certain point (equivalently, the queue that should be visited by the server at that point) may be dependent on the levels of external demand for each product type and is thus better modelled by a random environment than a round-robin assumption.

Aside from the characterisation as a polling model, the model that we study in the second part of this dissertation is also naturally characterised as a layered queueing network; see Figure 1.2. This is perhaps best explained in the setting of wireless random-access networks as given above. The nodes of that network can be interpreted as servers of the first layer, as they transmit the packets waiting for transmission. At the same time, they are also customers of the second layer, as they incur a delay before they activate to execute their transmission tasks. It goes without saying that this dual role of the nodes

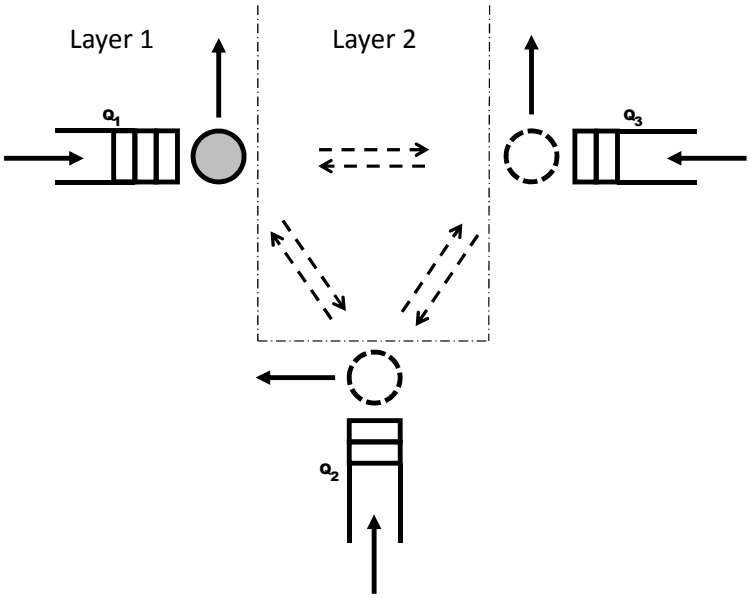

FIGURE 1.2: The Markovian polling model.

introduces significant interaction between the layers. For instance, if a node cannot activate due to congestion of the medium in the second layer, the number of packets to be sent by the corresponding node in the first layer builds up, increasing their delay in transmission. These interactions make analysis of the model non-trivial, especially since the nodes activate in a random order.

For this model, we analyse the waiting times of the first-layer customers in detail while taking the dynamics of the second layer directly into account. We also study the problem of how the implications of this analysis can be implemented in the wireless random-access networks setting, e.g. to achieve optimal back-off rates. Although the nodes are cooperative and strive for the common goal of minimising the overall packet delay, they operate autonomously and only have partial information available to them. As the remaining information needed to determine the optimal back-off rate can only be inferred from observed durations of periods between two transmissions in the medium so far, this is a non-trivial problem.

1.3.3 The carousel storage model

The third layered queueing model that we study also constitutes a polling model, but differs substantially from the Markovian polling model. More specifically, the third model involves one server visiting multiple service stations in a certain order like before, but each time he only serves at most one customer. Furthermore, at each station there is an infinite queue of customers that needs service. Before going through a service phase A at the server, a customer must first undergo a preparation phase B. Thus the server, after having finished serving a customer at one station, may have to wait for the preparation

phase of the customer at the next station to be completed. Immediately after the server concludes his service at some station, another customer from the infinite queue begins his preparation phase there.

This model finds countless applications in systems, where the order of service of the customers is important. For example, a typical operating strategy in healthcare clinics is to have a specialist rotate among several treatment rooms. In that case, the preparation phase represents the preliminary service a patient typically receives from an assistant or a nurse. The model, however, originates from warehousing. It was introduced in [188], where a storage facility is considered with bi-directional carousels and a picker that serves the carousels in turns. Therefore, we call this model the carousel storage model. The preparation phase represents the rotation time the carousel needs to bring the item to the origin, and the service time is the actual picking time. In that paper, the authors study the case of two carousels under specific assumptions. Later on, this special case for two stations has been further analysed under general distributional assumptions in [264]. For an extensive literature review on carousel systems, we refer to [167]. We will generalise many results found in [264] from two stations to multiple stations. This extension leads to significant challenges in the analysis, but provides valuable managerial insights.

Little work has been done on multiple-carousel warehouse systems. Multiple-carousel problems differ intrinsically from single-carousel problems in a number of ways. Such systems tend to be more complicated. The system cannot be viewed as a number of independently operating carousels [175], since the separate carousels interact by means of the picker that is assigned to them. Almost all studies involving systems with more than two carousels resort to simulation.

As mentioned above, the carousel storage model can be viewed as a polling model. In particular, it can be interpreted as an extension of a one-limited polling-type system (cf. [50, 88, 259]). In general, polling models with a k-limited service discipline (i.e. at most k customers are served per visit) are notoriously difficult to analyse, as their queue lengths do not allow for an interpretation as a multi-type branching process with immigration as explained in Section 1.3.2; see e.g. [208]. In our case, we also have the difficulty of an additional preparation phase before the actual service phase. We assume that when the service of a customer at a station ends, there is always a new customer waiting in front of the same station. In the carousel setting, this means that there is always an ample supply of items to pick from. Furthermore, in many service systems, appointments with customers occur on a scheduled basis, so that this assumption is also a natural one in that setting. As a result of this assumption, the analysis of the model is parallel to the study of the server in a one-limited polling-type system where each of the queues is critically loaded. Note that our main interest for this model is in the waiting time of the *server* rather than that of the *customers*.

This model is evidently a layered queueing network; see Figure 1.3. One may view the preparation time of a customer as a first phase of service. The service station (first station) acts in this case as a server of the first layer. However, the second phase of service (the actual operation) does not necessarily follow immediately. The service station might have to 'wait' for the server to finish working on other stations. At this stage, the service stations act as customers waiting to be served by the second layer, the server. Thus, we see that each service station acts both as a *server* (preparing the customer) and as a *customer* (waiting until the server completes his tasks in the previous stations). Apart from the waiting phases incurred by the service stations, however, interaction between the two

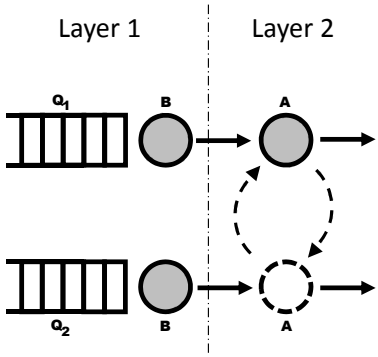

FIGURE 1.3: The carousel storage model.

layers also clearly manifests itself since the durations of the preparation phases in the first layer dictate how long the server of the second layer has to wait before a phase of service can start. In other words, the server of the second layer may also be interpreted as a customer of the first layer. Summarising, an important and distinguishing feature of this model is that there are multiple types of 'dual-role entities'. Not only are there first-layer servers which are customers in the second layer as in the two previous models, but now the second-layer server may be interpreted as a first-layer customer as well.

In the analysis that follows for this model, we initially assume that the server polls the service stations in a fixed, cyclic order. We also investigate a model variation where the server always serves the customer with the earliest completed preparation phase. Note that this 'dynamic' model variation almost completely reduces the carousel storage model to the extended machine repair model as described in Section 1.3.1 with a first-come-first-served repair policy, when interpreting the service stations and the server as the machines and the repairman, respectively. However, there are two fundamental differences. Apart from the fact that the first-layer queues are now assumed to consist of an infinite number of waiting customers, the focus of our analysis of this model lies on the waiting times of the second-layer server (or equivalently, the idle times of the repairman).

1.4 Contributions and overview of the thesis

In this section, we provide an overview of the results presented in the remainder of this thesis, along with their implications. For each of the two-layered queueing networks described in Section 1.3, we present an in-depth analysis of the relevant performance measures involved using a wide array of mathematical methods. At times, we will also turn to the question of how to allocate resources so as to optimise these performance measures as much as possible.

Whenever tractable, we will perform the analysis in an exact fashion. However, an exact analysis is often prohibited by the existing interactions between the different layers. These interactions are of a complicated nature, but they cannot be ignored due to the fact that they have a significant impact on the system. To overcome this problem, one may

resort to the application of numerical procedures. However, these methods are usually not transparent, do not give any insights into the nature of the interactions' impacts and are computationally complex. Hence, when an exact analysis is out of reach, we often aim for the derivation of symbolic approximations that are not only accurate and relatively easily implemented, but also show the main effects that the model parameters have on the performance measures. This ensures that the results obtained are not only suitable for optimalisation purposes, but that they may also provide insights into the actual effects of the interations between the layers on the system's performance.

Although this thesis is comprised of results for models that consist of two layers, the observations made may carry over to models with more than two layers. Also, the methods used to derive these results may be used as a starting point to analyse the many-layer setting.

We now sketch the organisation and give an overview of the main results that we obtain in the remaining chapters of this thesis. The thesis is divided in three parts, each of which is concerned with one of the layered queueing networks described in Section 1.3. We discuss each of these parts below, and we end with a note on some notational conventions used throughout the thesis.

Part I: The extended machine repair model Chapters 2–6 constitute the first part of this dissertation and contain our work on the extended machine repair model.

In particular, Chapter 2 shows how to compute the stationary distribution of performance measures in the extended machine repair model by applying the power-series algorithm. Although this is an algorithm geared for the numerical computation of stationary distributions, we run this algorithm in a symbolic fashion. This unconventional application of the power-series algorithm results in expressions that describe the behaviour of performance measures such as the mean queue length in the so-called *light-traffic* regime. This is the asymptotic regime where the utilisation rate of the servers approaches zero.

In Chapter 3, we study the behaviour of the performance measures in the *heavy-traffic* case, where the utilisation rates of the servers are such that the queues are on the verge of instability. Instability of a queue occurs when the amount of work that the server can handle per time unit does not exceed the amount of work per time unit that is brought to the server by arriving customers. In such a case, the queue will grow indefinitely without bound. By combining a classical functional central limit theorem approach with matrix-analytic methods, we obtain heavy-traffic results for networks of parallel single-server queues where the service speeds of all servers are modulated by a single continuous-time Markov chain. This model covers the extended machine repair model, but it is actually much broader. As a consequence, the results of this chapter are not restricted to the extended machine repair model.

Chapter 4 combines the light-traffic and the heavy-traffic results from Chapters 2 and 3, respectively, to obtain approximations of the mean queue lengths of the first-layer queues in the extended machine repair model. The approximations that we obtain are in closed form. Furthermore, numerical results show that these approximations are highly accurate. As a result, they can be used for optimisation purposes.

In Chapter 5, we obtain approximations for the *complete* (marginal) queue length distributions of the first-layer queues in the extended machine repair model. We do this by drawing a connection between a first-layer queue and a single-server queue with server

vacations that are mutually (one-)dependent and closely resemble the correlated downtimes of a machine. We obtain an approximate expression for the queue length distribution of the latter queue in terms of probability generating functions and use this expression as an approximation for the queue length distribution of a first-layer queue in the setting of the extended machine repair model. Although this approximation does not perform as well as the approximations in Chapter 4 when approximating the mean queue lengths, numerical results nonetheless show that it is reasonably accurate over a wide range of parameter settings. Furthermore, this approximation can also be used to approximate variances or tail probabilities of the queue length distributions.

Part I is concluded by Chapter 6, where we concern ourselves with the dynamic optimisation problem of how to allocate the resources of the repairman so as to minimise a weighted average of the mean queue lengths of the first-layer queues. We derive several structural properties of the repairman's optimal policy. As the actual optimal policy is hard to find analytically, we also derive a near-optimal policy by combining results from queueing theory with techniques from Markov decision theory.

The results found in Chapter 2 are largely based on [P8] and the results in Chapter 3 stem from [P11, P12]. The approximations derived in Chapter 4 have also been discussed in [P8]. Finally, Chapters 5 and 6 are based on the results of [P6] and [P3], respectively.

Part II: The Markovian polling model In the second part of this thesis, which is comprised of Chapters 7–9, we provide an analysis of the Markovian polling model.

In Chapter 7, we derive exact expressions for the probability generating functions of the marginal queue length distributions under the assumptions that there are only two queues and that the server initiates a switch-over period only when there are no customers waiting in the queue he is currently visiting (so-called *exhaustive service*). Furthermore, we obtain explicit expressions for the (properly scaled) queue length distribution in a heavy-traffic regime (as before, the case where the server is presented with a critical load). It turns out that in this regime, the waiting-time and queue length distributions are very similar to those encountered in a regular cyclic polling system.

Chapter 8 concerns itself with general Markovian polling systems that do not necessarily satisfy the two-queue assumption or the exhaustive-service assumption made in Chapter 7. This considerably complicates the analysis, since without these assumptions, the joint queue length process of the Markovian polling system observed at polling epochs cannot be modelled as a multi-type branching process with immigration as described in Section 1.3.2. Nevertheless, by exploiting a functional equation for the (probability generating function of the) joint queue length distribution at points in time at which the server starts a visit period, we show how to derive expressions for the (cross-)moments of the queue lengths. We also derive a pseudo-conservation law, from which an expression for the stationary expected amount of (waiting) work present in the system follows.

In Chapter 9, we turn to the question of how certain parameters of the model should be chosen so as to minimise a (possibly weighted) average of the mean queue lengths. We also focus on the application to wireless random-access networks as given in Section 1.3.2. In particular, we show how the optimisation results could be implemented in these networks while dealing with the issues caused by their decentralised character.

The results presented in Chapter 7 can be found in [P5]. Chapters 8 and 9 are largely built on the results of [P4].

Part III: The carousel storage model Chapters 10 and 11 together form the final part of the thesis, where we perform a detailed analysis of the carousel storage model. In particular, we analyse both the transient and the long-run probabilistic behaviour of this model by quantifying the waiting-time distribution of the server in the second layer, which is directly connected to the system's efficiency and throughput.

In Chapter 10, we consider the carousel storage model under the additional assumption that the server polls the stations in a cyclic order. We give a sufficient condition for the existence of a limiting distribution for the waiting time of the server, and we study the tail behaviour of this distribution. We also show that if the preparation times are exponentially distributed, the waiting time of the server is also exponentially distributed with the same rate, provided it is non-zero. We subsequently compute the probability of a non-zero waiting time by combining the memoryless property of the exponential distribution with the analysis of the appropriate discrete-time Markov chain. Finally, this chapter provides extensive numerical results that identify the main effects of the model parameters on the waiting times of the server.

In Chapter 11, we study the question of how the waiting-time distribution of the server is affected if we drop the restriction that the server is required to serve the service stations in a cyclic manner. Although the waiting-time distributions corresponding to the cyclic and the non-cyclic cases are not necessarily stochastically ordered, we prove that the mean waiting time of the server in the non-cyclic case never exceeds the mean waiting time in the cyclic case. We also investigate numerically how the earlier discovered main effects of the model parameters are affected when dropping the assumption of cyclic service.

The work of Chapter 10 is based on [P13]. The results of Chapter 11 can be found in [P7, P13].

Notational conventions We end this chapter by introducing several notational conventions. Unless otherwise stated, the notation in all chapters adheres to the following. Throughout the thesis, vectors are printed in bold face. The vectors $\mathbf{0}$ and $\mathbf{1}$ represent vectors of appropriate size of which each element equals zero and one, respectively. The vector $\boldsymbol{e}_j$ represents a unit vector of appropriate size of which the j-th entry equals one and all other entries equal zero. Furthermore, we denote the indicator function on the event A by $\mathbb{1}_{\{A\}}$. The symbols $\wedge$ and $\vee$ represent a logical conjunction and a logical disjunction, respectively, and equality in distribution is denoted by $\stackrel{d}{=}$. We also use $(x)^-$ and $(x)^+$ as shorthand notation for $\min\{x,0\}$ and $\max\{x,0\}$, respectively.

The Laplace-Stieltjes transform of (the distribution of) any continuous random variable U is denoted by $\widetilde{U}(s) = \mathbb{E}[e^{-sU}]$ and is defined for $\Re(s) \geq 0$. Likewise, for any discrete random variable X or any n-dimensional vector of discrete random variables $\boldsymbol{Y} = (Y_1, \ldots, Y_n)$, the one-dimensional probability generating function $\widetilde{X}(z_1) = \mathbb{E}[z_1^X]$ and the n-dimensional probability generating function $\widetilde{Y}(z_1, \ldots, z_n) = \mathbb{E}[\prod_{k=1}^N z_k^{Y_k}]$ are defined for any complex-valued $z_1, \ldots, z_N$ for which $|z_1|, \ldots, |z_n|$ do not exceed one.

Part I

The extended machine repair model

2

NUMERICAL COMPUTATION AND LIGHT-TRAFFIC ASYMPTOTICS

In this chapter, we apply the power-series algorithm to the extended machine repair model as introduced in Section 1.3.1. This algorithm provides a powerful means of numerically computing performance measures such as the moments of the queue length distribution of the first-layer queues. However, it also allows one to gain insight into the so-called light-traffic behaviour of these performance measures. In other words, one can derive the behaviour of the performance measures with respect to the utilisation rate of the machines in terms of symbolic expressions in case the utilisation rate approaches zero. The light-traffic insights gained in this chapter will act as one of the building blocks for the approximations that we derive in Chapter 4 for the mean queue lengths of the queues of products.

2.1 Introduction

This chapter considers the power-series algorithm and its application to the extended machine repair model. The power-series algorithm is a numerical algorithm used to compute the steady-state distribution of multi-dimensional queueing systems. Although it may be trivial to derive the global balance equations for these systems, they usually cannot be solved recursively due to a lack of a product-form solution. The basic idea behind the power-series algorithm is the transformation of the non-recursively solvable set of balance equations into a recursively solvable set of equations by adding one dimension to the state space. This is achieved by expressing the steady-state probabilities as power series in some variable in light traffic, which allows calculation of steady-state probabilities. The idea behind this algorithm stems from Hooghiemstra et al. [126] and has been further developed by Blanc (see e.g. [40, 41]). For an overview of the power-series algorithm and its initial literature, see [42, 146].

The use of the power-series algorithm is in many regards advantageous over numerical methods such as simulation. The computation time needed to achieve accurate numerical results is generally much less, especially for lightly loaded systems. Apart from this, the computational scheme provided by the power-series algorithm can also be executed symbolically to compute the light-traffic behaviour of several performance measures per-

taining to the first-layer queues, i.e. the behaviour in case the load offered to the machines tends to zero.

In Section 2.2, we formulate the model assumptions and the notation required to apply the power-series algorithm to the extended machine repair model. Then, we explain how to implement the power-series algorithm for this model in Section 2.3. Finally, we derive symbolic expressions that shed light on the light-traffic behaviour of the first-layer queues in Section 2.4.

2.2 Model description and notation

In this section, we state our model assumptions and we introduce the notation that we use in this chapter to analyse the extended machine repair model as depicted in Figure 1.1. The first layer of this model consists of two machines M_1 and M_2 as well as the corresponding queues Q_1 and Q_2, which we will also refer to as first-layer queues. Products arrive at Q_i according to a Poisson process with rate λ_i. The service requirement of a product in Q_i is exponentially distributed with parameter μ_i. We denote the load offered to Q_i by $\rho_i = \frac{\lambda_i}{\mu_i}$. The steady-state queue length of Q_i, including the product in service, is denoted by L_i. Furthermore, the time between the arrival of a type-i product and the end of its service is referred to as the sojourn time S_i. After an exponentially (σ_i) distributed uptime, denoted by U_i, the machine M_i serving Q_i will break down, and the service of Q_i stops. The service of a product in progress is then aborted and will be restarted once the machine is operational again. When a machine breaks down, it moves to the repair queue, where it will wait if the repairman is busy repairing the other machine. Otherwise, the repair will start immediately. Thus, a downtime D_i of a machine M_i consists of a repair time and *possibly* a waiting time. The time R_i needed for a repairman to return M_i to an operational state is exponentially (ν_i) distributed. After a repair, the machine returns to Q_i and commences service again. All interarrival, service, uptime and repair times are assumed to be independent.

In various computations, we need to keep track of the state of the background environment, namely whether the two machines are working or not. To this end, let $\{\mathbf{\Phi}(t), t \geq 0\}$ be the continuous-time Markov chain describing the state of the machines M_1 and M_2. More specifically, $\mathbf{\Phi}(t) = (\Phi_1(t), \Phi_2(t))$ specifies for each machine whether it is up (U), in repair (R) or waiting for repair (W) at time t. This Markov chain operates on the state space $\mathscr{S} = \{(U,U),(U,R),(R,U),(W,R),(R,W)\}$ with generator matrix Q. Its stationary distribution vector $\boldsymbol{\pi} = (\pi_i)_{i\in\mathscr{S}}$ is uniquely determined by the equations $\boldsymbol{\pi}Q = \mathbf{0}$ and $\sum_{j\in\mathscr{S}} \pi_j = 1$.

The queue length of a first-layer queue depends heavily on the availability of its machine in the past. To keep track of the latter, let $C_i(t)$ represent the amount of time the machine M_i has been working in the time period $[0,t)$. Assuming the process $\{\mathbf{\Phi}(t), t \geq 0\}$ is already in stationarity at $t = 0$, $C_i(t)$ is defined as

$$C_i(t) = \int_{s=0}^{t} \mathbb{1}_{\{\Phi_i(s)=U\}} ds. \tag{2.1}$$

The long-run time-averaged mean of the process $\{C_i(t), t \geq 0\}$, i.e. the fraction of time

M_i is up, is given by

$$m_{C,i} = \lim_{t\to\infty} \frac{\mathbb{E}[C_i(t)]}{t} = \lim_{t\to\infty} \frac{\int_{s=0}^{t} \mathbb{P}(\Phi_i(s) = U)ds}{t} = \sum_{\varphi\in\{\psi:\psi_i=U\wedge\psi\in\mathscr{S}\}} \pi_\varphi.$$

Note that by standard renewal arguments, we also have that $m_{C,i} = \frac{\mathbb{E}[U_i]}{\mathbb{E}[U_i]+\mathbb{E}[D_i]}$. To keep track of the level of saturation of Q_i, we introduce the notion of *normalised load*. If M_i never breaks down, the stability condition for Q_i reads $\rho_i < 1$. However, in the case of breakdowns, this condition is not sufficient any longer, as M_i only works for a fraction $m_{C,i}$ of the time. We therefore define $\hat{\rho}_i = \frac{\rho_i}{m_{C,i}}$. We also refer to $\hat{\rho}_i$ as the normalised load of Q_i. Taking the breakdowns of M_i into account, the stability condition for Q_i thus reads $\hat{\rho}_i < 1$.

Throughout this chapter, we denote the L^1-norm of a vector $\boldsymbol{z}$ consisting of n elements by $|\boldsymbol{z}| = z_1 + \cdots + z_n$. Finally, for two functions $f(x)$ and $g(x)$, we write $f(x) = \mathcal{O}(g(x))$ if $\lim_{x\downarrow 0} |f(x)/g(x)| < \infty$.

2.3 Application of the power-series algorithm

In this section, we show how the power-series algorithm can be used to analyse the extended machine repair model. The power-series algorithm is typically used to compute the steady-state distribution of several classes of multiple-queue systems, e.g. those which fit in the class of quasi birth-and-death processes. The extended machine repair model is such a multi-dimensional quasi birth-and-death process and consists of two components. The first component $\{\boldsymbol{L}(t) = (L_1(t), L_2(t)), t \geq 0\}$ describes the queue length at each of the queues. The second component models any non-exponentiality in the system. In our system, non-exponentiality is caused by the fact that the machines alternate between uptimes and downtimes and is represented by the process $\{\boldsymbol{\Phi}(t), t \geq 0\}$. In this way, $\{(\boldsymbol{L}(t), \boldsymbol{\Phi}(t)), t \geq 0\}$ can be seen as a continuous-time Markov chain on the state space $\mathbb{N}^2 \times \mathscr{S}$. When the system is stable, the steady-state probabilities $p(\boldsymbol{l}, \varphi), (\boldsymbol{l}, \varphi) \in \mathbb{N}^2 \times \mathscr{S}$, can be obtained in principle by solving the set of global balance equations. This, however, is not a trivial task, as the set of equations is not recursively solvable. To overcome this problem, we apply the power-series algorithm. As a result, performance measures of the form $\mathbb{E}[g(\boldsymbol{L}, \boldsymbol{\Phi})]$ can be computed, where $g(\cdot)$ is an arbitrary function and $(\boldsymbol{L}, \boldsymbol{\Phi}) = \lim_{t\to\infty}(\boldsymbol{L}(t), \boldsymbol{\Phi}(t))$. We first define the one-step transition rates and the global balance equations corresponding to the Markov chain $\{(\boldsymbol{L}(t), \boldsymbol{\Phi}(t)), t \geq 0\}$ in Section 2.3.1. Then, we apply the power-series algorithm directly to the extended machine repair model in Section 2.3.2.

2.3.1 Preliminaries

We first study the continuous-time Markov chain $\{(\boldsymbol{L}(t), \boldsymbol{\Phi}(t)), t \geq 0\}$ and consider its one-step transition rates and global balance equations. The one-step transition rate corresponding to the transition from state $(\boldsymbol{l}, \varphi) \in \mathbb{N}^2 \times \mathscr{S}$ to state $(\boldsymbol{l}+\boldsymbol{e}_i, \varphi)$ equals the arrival rate λ_i. However, in order to fully exploit the flexibility that the power-series algorithm provides, we parameterise each of the arrival rates by a 'relative' arrival rate $a^{(i)}(\boldsymbol{l}, \varphi)$

times a parameter χ. The quantity χ will be used by the power-series algorithm to introduce another dimension to the state space. For $(\boldsymbol{l},\boldsymbol{\varphi}) \in \mathbb{N}^2 \times \mathscr{S}$ and $\boldsymbol{\psi} \in \mathscr{S}$, we define the one-step transition rates as follows:

$\chi a^{(j)}(\boldsymbol{l},\boldsymbol{\varphi})$: the arrival rate at Q_j at state $(\boldsymbol{l},\boldsymbol{\varphi})$, leading to a transition to state $(\boldsymbol{l}+\boldsymbol{e}_j,\boldsymbol{\varphi})$, $j=1,2$,

$d^{(j)}(\boldsymbol{l},\boldsymbol{\varphi})$: the departure rate from Q_j at state $(\boldsymbol{l},\boldsymbol{\varphi})$, leading to a transition to state $(\boldsymbol{l}-\boldsymbol{e}_j,\boldsymbol{\varphi})$, with $d^{(j)}(\boldsymbol{l},\boldsymbol{\varphi})=0$ if $l_j=0$, $j=1,2$,

$u(\boldsymbol{l},\boldsymbol{\varphi},\boldsymbol{\psi})$: the transition rate from $(\boldsymbol{l},\boldsymbol{\varphi})$ to $(\boldsymbol{l},\boldsymbol{\psi})$.

Linking this with the notation given in Section 2.2, this means that for $j=1,2$ and $(\boldsymbol{l},\boldsymbol{\varphi}) \in \mathbb{N}^2 \times \mathscr{S}$:

$$
\begin{aligned}
&\chi a^{(j)}(\boldsymbol{l},\boldsymbol{\varphi}) = \lambda_j,\\
&d^{(j)}(\boldsymbol{l},\boldsymbol{\varphi}) = \mu_j \mathbb{1}_{\{l_j>0\}}\mathbb{1}_{\{\varphi_j=U\}},\\
&u(\boldsymbol{l},(U,U),(R,U)) = u(\boldsymbol{l},(U,R),(W,R)) = \sigma_1,\\
&u(\boldsymbol{l},(U,U),(U,R)) = u(\boldsymbol{l},(R,U),(R,W)) = \sigma_2,\\
&u(\boldsymbol{l},(R,U),(U,U)) = u(\boldsymbol{l},(R,W),(U,R)) = \nu_1,\\
&u(\boldsymbol{l},(U,R),(U,U)) = u(\boldsymbol{l},(W,R),(R,U)) = \nu_2.
\end{aligned}
$$

It remains to choose an appropriate value for χ. For the application of the power-series algorithm, it is generally required that there exists a positive real χ^* such that both Q_1 and Q_2 are stable for $0 \le \chi < \chi^*$. To satisfy this requirement, we choose

$$
\chi = \hat{\rho}_1 = \frac{\lambda_1}{\mu_1 m_{C,1}}. \tag{2.2}
$$

This leads to $a^{(1)}(\boldsymbol{l},\boldsymbol{\varphi}) = \mu_1 m_{C,1}$ and $a^{(2)}(\boldsymbol{l},\boldsymbol{\varphi}) = \frac{\lambda_2}{\lambda_1}\mu_1 m_{C,1}$. Note that for the current choice of χ, there indeed exists an upper bound below which both queues are stable. Evidently, when the normalised load does not exceed one, Q_1 is stable. Moreover, the ratio between the service rates μ_1 and μ_2, as well as the ratio between the time fractions $m_{C,1}$ and $m_{C,2}$, is assumed to be finite; i.e. we assume that none of the service rates and time fractions are zero. Thus, there must exist a positive real c such that Q_2 is stable whenever $0 \le \chi < c$. As a result, the requirement is satisfied when taking $\chi^* = \min\{1, c\}$.

The global balance equations of the Markov chain $\{(\boldsymbol{L}(t),\boldsymbol{\Phi}(t)), t \ge 0\}$, expressed in the steady-state probabilities $p(\boldsymbol{l},\boldsymbol{\varphi})$, are given by

$$
\begin{aligned}
&\left(\sum_{j=1}^{2}\left(\chi a^{(j)}(\boldsymbol{l},\boldsymbol{\varphi}) + d^{(j)}(\boldsymbol{l},\boldsymbol{\varphi})\right) + \sum_{\boldsymbol{\psi}\in\mathscr{S}} u(\boldsymbol{l},\boldsymbol{\varphi},\boldsymbol{\psi})\right) p(\boldsymbol{l},\boldsymbol{\varphi})\\
&= \chi \sum_{j=1}^{2} a^{(j)}(\boldsymbol{l}-\boldsymbol{e}_j,\boldsymbol{\varphi}) p(\boldsymbol{l}-\boldsymbol{e}_j,\boldsymbol{\varphi}) \mathbb{1}_{\{l_j>0\}} + \sum_{j=1}^{2} d^{(j)}(\boldsymbol{l}+\boldsymbol{e}_j,\boldsymbol{\varphi}) p(\boldsymbol{l}+\boldsymbol{e}_j,\boldsymbol{\varphi})\\
&\quad + \sum_{\boldsymbol{\psi}\in\mathscr{S}} u(\boldsymbol{l},\boldsymbol{\psi},\boldsymbol{\varphi}) p(\boldsymbol{l},\boldsymbol{\psi})
\end{aligned} \tag{2.3}
$$

for any $(\boldsymbol{l},\boldsymbol{\varphi}) \in \mathbb{N}^2 \times \mathscr{S}$. We also have the normalisation equation

$$\sum_{(\boldsymbol{l},\boldsymbol{\varphi})\in\mathbb{N}^2\times\mathscr{S}} p(\boldsymbol{l},\boldsymbol{\varphi}) = 1. \tag{2.4}$$

To substitute the steady-state probabilities, we use the following property (cf. [146, 247]).

PROPERTY 2.3.1. For each state $(\boldsymbol{l},\boldsymbol{\varphi})$, it holds that $p(\boldsymbol{l},\boldsymbol{\varphi}) = \mathscr{O}(\chi^{|\boldsymbol{l}|})$. This property is valid for any quasi birth-and-death process where for each state $(\boldsymbol{l},\boldsymbol{\varphi})$ with $\boldsymbol{l} \neq \mathbf{0}$, either $p(\boldsymbol{l},\boldsymbol{\varphi}) = 0$, or there exists a path $\boldsymbol{\varphi}^{(0)},\boldsymbol{\varphi}^{(1)},\ldots,\boldsymbol{\varphi}^{(\zeta)}$ in $\mathscr{S}$ for some $\zeta \in \{0,\ldots,|\mathscr{S}|\}$ such that

$$\boldsymbol{\varphi}^{(0)} = \boldsymbol{\varphi},\ u(\boldsymbol{l},\boldsymbol{\varphi}^{(i-1)},\boldsymbol{\varphi}^{(i)}) > 0$$

for $i \in \{1,\ldots,\zeta\}$ and there is at least one queue with a non-zero departure rate in the state $(\boldsymbol{l},\boldsymbol{\varphi}^{(\zeta)})$.

Note that the conditions mentioned in Property 2.3.1 are obviously met in the extended machine repair model. For any queue length configuration, there exists a path from any $\boldsymbol{\varphi} \in \mathscr{S}$ to the auxiliary state (U,U). In this state, both machines are operational and departure rates for both of the queues are non-zero. We therefore introduce the power-series expansion

$$p(\boldsymbol{l},\boldsymbol{\varphi}) = \chi^{|\boldsymbol{l}|}\sum_{k=0}^{\infty}\chi^k b(k;\boldsymbol{l},\boldsymbol{\varphi}) \tag{2.5}$$

for the steady-state probabilities corresponding to the states $(\boldsymbol{l},\boldsymbol{\varphi}) \in \mathbb{N}^2 \times \mathscr{S}$. The coefficients $b(k;\boldsymbol{l},\boldsymbol{\varphi})$ appearing in (2.5) are still unknown. We focus on the computation of these coefficients in the next section.

2.3.2 Computational scheme

We now apply the power-series algorithm to the extended machine repair model and derive a recursive, computational scheme for this model. We obtain and solve a recursive set of equations for the coefficients $b(k;\boldsymbol{l},\boldsymbol{\varphi})$ defined in (2.5). From this, all steady-state probabilities can be computed as well as any performance measures derived from them. We first substitute the power-series expansion (2.5) into the balance equations given in (2.3). This leads to a polynomial expression in χ for both sides of the equations. By equating corresponding powers of χ, we obtain a recursion in the coefficients $b(k;\boldsymbol{l},\boldsymbol{\varphi})$ for $k \in \mathbb{N}, (\boldsymbol{l},\boldsymbol{\varphi}) \in \mathbb{N}^2 \times \mathscr{S}$. As a result, we can compute many performance measures by writing them as a power series in χ with different coefficients, but still involving the obtained values for $b(k;\boldsymbol{l},\boldsymbol{\varphi})$ for $k \in \mathbb{N}, (\boldsymbol{l},\boldsymbol{\varphi}) \in \mathbb{N}^2 \times \mathscr{S}$.

As mentioned above, the first step of the power-series algorithm constitutes the substitution of the power-series expansion (2.5) into the balance equations given in (2.3), which results in the following set of equations for the coefficients $b(k;\boldsymbol{l},\boldsymbol{\varphi})$:

$$\begin{aligned}&\chi^{|\boldsymbol{l}|}\sum_{k=0}^{\infty}\chi^k\left(\sum_{j=1}^{2}\left(\chi a^{(j)}(\boldsymbol{l},\boldsymbol{\varphi}) + d^{(j)}(\boldsymbol{l},\boldsymbol{\varphi})\right) + \sum_{\boldsymbol{\psi}\in\mathscr{S}} u(\boldsymbol{l},\boldsymbol{\varphi},\boldsymbol{\psi})\right) b(k;\boldsymbol{l},\boldsymbol{\varphi})\\ &= \chi^{|\boldsymbol{l}|-1}\sum_{k=0}^{\infty}\chi^k\sum_{j=1}^{2}\chi a^{(j)}(\boldsymbol{l}-\boldsymbol{e}_j,\boldsymbol{\varphi}) b(k;\boldsymbol{l}-\boldsymbol{e}_j,\boldsymbol{\varphi})\mathbb{1}_{\{\boldsymbol{l}_j>0\}}\end{aligned}$$

$$+\chi^{|\boldsymbol{l}|+1}\sum_{k=0}^{\infty}\chi^k\sum_{j=1}^{2}d^{(j)}(\boldsymbol{l}+\boldsymbol{e}_j,\boldsymbol{\varphi})b(k;\boldsymbol{l}+\boldsymbol{e}_j,\boldsymbol{\varphi})$$
$$+\chi^{|\boldsymbol{l}|}\sum_{k=0}^{\infty}\chi^k\sum_{\boldsymbol{\psi}\in\mathscr{S}}u(\boldsymbol{l},\boldsymbol{\psi},\boldsymbol{\varphi})b(k;\boldsymbol{l},\boldsymbol{\psi})$$

for any $(\boldsymbol{l},\boldsymbol{\varphi})\in\mathbb{N}^2\times\mathscr{S}$. After eliminating the factor $\chi^{|\boldsymbol{l}|}$ from both sides of this set of equations, we obtain a polynomial equation of the form $\sum_{i=0}^{\infty}c_i\chi^i=\sum_{i=0}^{\infty}\gamma_i\chi^i$. Since this equation holds for every $\chi\in[0,\chi^*)$, the coefficients of corresponding powers of χ are equal. Thus, we have that $c_i=\gamma_i$ for all i, which leads to

$$\begin{aligned}&\left(\sum_{j=1}^{2}d^{(j)}(\boldsymbol{l},\boldsymbol{\varphi})+\sum_{\boldsymbol{\psi}\in\mathscr{S}}u(\boldsymbol{l},\boldsymbol{\varphi},\boldsymbol{\psi})\right)b(k;\boldsymbol{l},\boldsymbol{\varphi})\\&=\sum_{j=1}^{2}a^{(j)}(\boldsymbol{l}-\boldsymbol{e}_j,\boldsymbol{\varphi})b(k;\boldsymbol{l}-\boldsymbol{e}_j,\boldsymbol{\varphi})\mathbb{1}_{\{\boldsymbol{l}_j>0\}}-\sum_{j=1}^{2}a^{(j)}(\boldsymbol{l},\boldsymbol{\varphi})b(k-1;\boldsymbol{l},\boldsymbol{\varphi})\mathbb{1}_{\{k>0\}}\\&\quad+\sum_{j=1}^{2}d^{(j)}(\boldsymbol{l}+\boldsymbol{e}_j,\boldsymbol{\varphi})b(k-1;\boldsymbol{l}+\boldsymbol{e}_j,\boldsymbol{\varphi})\mathbb{1}_{\{k>0\}}\\&\quad+\sum_{\boldsymbol{\psi}\in\mathscr{S}}u(\boldsymbol{l},\boldsymbol{\psi},\boldsymbol{\varphi})b(k;\boldsymbol{l},\boldsymbol{\psi})\end{aligned}\tag{2.6}$$

for each $(k;\boldsymbol{l},\boldsymbol{\varphi})\in\mathbb{N}^3\times\mathscr{S}$. The resulting set of equations now forms a recursive scheme with respect to the partial ordering $\prec$ of the vectors $(k;\boldsymbol{l},\boldsymbol{\varphi})$, where $(k;\boldsymbol{l},\boldsymbol{\varphi})\prec(\widehat{k};\widehat{\boldsymbol{l}},\widehat{\boldsymbol{\varphi}})$ if

$$\left[k+|\boldsymbol{l}|<\widehat{k}+|\widehat{\boldsymbol{l}}|\right]\text{ or }\left[k+|\boldsymbol{l}|=\widehat{k}+|\widehat{\boldsymbol{l}}|\wedge k<\widehat{k}\right].$$

Indeed, we see that (2.6) expresses the coefficients $b(k;\boldsymbol{l},\boldsymbol{\varphi})$ in terms of coefficients of lower order than $(k;\boldsymbol{l},\boldsymbol{\varphi})$ with respect to $\prec$, except for the coefficient $b(k;\boldsymbol{l},\boldsymbol{\psi})$ in the last line. Therefore, the coefficients $b(k;\boldsymbol{l},\boldsymbol{\varphi})$ can be calculated recursively in increasing order with respect to $\prec$, where for each combination $(k;\boldsymbol{l})$ a set of at most $|\mathscr{S}|$ linear equations must be solved. This set of equations generally possesses a unique solution. The only exception is when the system is totally empty ($\boldsymbol{l}=\boldsymbol{0}$) and thus all departure rates vanish. For $\boldsymbol{l}=\boldsymbol{0},\boldsymbol{\varphi}\in\mathscr{S}$, the set of equations in (2.6) reduces to

$$\sum_{\boldsymbol{\psi}\in\mathscr{S}}u(\boldsymbol{0},\boldsymbol{\varphi},\boldsymbol{\psi})b(k;\boldsymbol{0},\boldsymbol{\varphi})=\sum_{\boldsymbol{\psi}\in\mathscr{S}}u(\boldsymbol{0},\boldsymbol{\psi},\boldsymbol{\varphi})b(k;\boldsymbol{0},\boldsymbol{\psi})+y(k;\boldsymbol{\varphi}),\tag{2.7}$$

where

$$y(k;\boldsymbol{\varphi})=-\sum_{j=1}^{2}a^{(j)}(\boldsymbol{0},\boldsymbol{\varphi})b(k-1;\boldsymbol{0},\boldsymbol{\varphi})\mathbb{1}_{\{k>0\}}+\sum_{j=1}^{2}d^{(j)}(\boldsymbol{e}_j,\boldsymbol{\varphi})b(k-1;\boldsymbol{e}_j,\boldsymbol{\varphi})\mathbb{1}_{\{k>0\}}.$$

By summing the equations of (2.7) over all $\boldsymbol{\varphi}\in\mathscr{S}$, we observe that these are dependent sets of equations for the coefficients $b(k;\boldsymbol{0},\boldsymbol{\varphi})$. The dependent sets are not contradictory, since we have that $\sum_{\boldsymbol{\varphi}\in\mathscr{S}}y(k;\boldsymbol{\varphi})=0$ due to a necessary balance between the empty states and the states with one product in the system. However, due to the dependence,

additional equations are needed. The law of total probability provides an additional equation between the coefficients $b(k;\boldsymbol{l},\boldsymbol{\varphi})$ for $(k;\boldsymbol{l},\boldsymbol{\varphi}) \in \mathbb{N}^3 \times \mathscr{S}$ when the system is empty. Namely, observe that if we take $\chi = 0$ in (2.5), which corresponds to zero arrival rates, all terms vanish, except for the one corresponding to $k = 0$. Thus, from the law of total probability (i.e. the normalisation equation (2.4)), we have

$$\sum_{\boldsymbol{\varphi}\in\mathscr{S}} b(0;\mathbf{0},\boldsymbol{\varphi}) = \sum_{\boldsymbol{\varphi}\in\mathscr{S}} p(\mathbf{0},\boldsymbol{\varphi}) = \sum_{(\boldsymbol{l},\boldsymbol{\varphi})\in\mathbb{N}^2\times\mathscr{S}} p(\boldsymbol{l},\boldsymbol{\varphi}) = 1, \tag{2.8}$$

where the first equality follows from (2.5). The second equality follows due to the fact that if all arrival rates are zero, then all $p(\boldsymbol{l},\boldsymbol{\varphi})$ for which $\boldsymbol{l} \neq \mathbf{0}$ are zero. Similarly, (2.4) implies for $k > 0$ that

$$\sum_{\boldsymbol{\varphi}\in\mathscr{S}} b(k;\mathbf{0},\boldsymbol{\varphi}) = -\sum_{0<|\boldsymbol{l}|\leq k} \sum_{\boldsymbol{\psi}\in\mathscr{S}} b(k-|\boldsymbol{l}|;\boldsymbol{l},\boldsymbol{\psi}). \tag{2.9}$$

To see how (2.9) is derived, we argue as follows. First, we substitute (2.5) into (2.4) and thus write the normalisation equation as a power series in χ. As this equation needs to hold true for each value of $\chi \in [0,\chi^*)$, the first-order and higher-order coefficients of this power series must be equal to zero. Based on this, we conclude that for every $k > 0$, it holds that $\sum_{0\leq|\boldsymbol{l}|\leq k}\sum_{\boldsymbol{\varphi}\in\mathscr{S}} b(k-|\boldsymbol{l}|;\boldsymbol{l},\boldsymbol{\varphi}) = 0$. Equation (2.9) now follows by moving terms for which $|\boldsymbol{l}| > 0$ to the right-hand side.

Note that the right-hand side of (2.9) consists of terms of lower order than $b(k;\mathbf{0},\boldsymbol{\varphi})$ with respect to $\prec$. All but one of the equations of (2.7) in combination with (2.8) or (2.9) determine $b(k;\mathbf{0},\boldsymbol{\varphi})$. In general, this set of equations has a unique solution if the process, conditioned on the event that both queues are empty and no arrivals occur at all, is irreducible on the subset of $\mathscr{S}$ of reachable states. This condition holds for the current model, as the continuous-time Markov chain $\{\boldsymbol{\Phi}(t), t \geq 0\}$ on the state space $\mathscr{S}$ is evidently irreducible.

One can now recursively compute all the coefficients $b(k;\boldsymbol{n},\boldsymbol{\varphi})$ for $k \in \mathbb{N}$, $(\boldsymbol{n},\boldsymbol{\varphi}) \in \mathbb{N}^2 \times \mathscr{S}$. This not only allows for the computation of the steady-state probabilities themselves, but also for the computation of any function of the steady-state probabilities. More specifically, let $g(\boldsymbol{l},\boldsymbol{\varphi})$ represent a function which maps values from the state space $\mathbb{N}^2\times\mathscr{S}$ to a real value. Most common performance measures, including moments of the queue lengths, can be expressed in the form $\mathbb{E}[g(\boldsymbol{L},\boldsymbol{\Phi})]$. Using (2.5), the expectation of $g(\boldsymbol{L},\boldsymbol{\Phi})$ is defined as

$$\mathbb{E}[g(\boldsymbol{L},\boldsymbol{\Phi})] = \sum_{(\boldsymbol{l},\boldsymbol{\varphi})\in\mathbb{N}^2\times\mathscr{S}} g(\boldsymbol{l},\boldsymbol{\varphi})p(\boldsymbol{l},\boldsymbol{\varphi}) = \sum_{m=0}^{\infty}\sum_{|\boldsymbol{l}|=m}\sum_{\boldsymbol{\varphi}\in\mathscr{S}} g(\boldsymbol{l},\boldsymbol{\varphi})\sum_{k=0}^{\infty}\chi^{k+m}b(k;\boldsymbol{l},\boldsymbol{\varphi}).$$

By changing the index of the last sum, substituting $k-m$ for k and subsequently changing the order of summation, we obtain

$$\mathbb{E}[g(\boldsymbol{L},\boldsymbol{\Phi})] = \sum_{k=0}^{\infty}\chi^k\sum_{m=0}^{k}\sum_{|\boldsymbol{l}|=m}\sum_{\boldsymbol{\varphi}\in\mathscr{S}} g(\boldsymbol{l},\boldsymbol{\varphi})b(k-m;\boldsymbol{l},\boldsymbol{\varphi}).$$

This implies that performance measures of the form $\mathbb{E}[g(\boldsymbol{L},\boldsymbol{\Phi})]$ can also be written as a power series in χ:

$$\mathbb{E}[g(\boldsymbol{L},\boldsymbol{\Phi})] = \sum_{k=0}^{\infty}\chi^k f(k), \tag{2.10}$$

where the coefficients are given by

$$f(k) = \sum_{0 \le |\boldsymbol{l}| \le k} \sum_{\boldsymbol{\varphi} \in \mathscr{S}} g(\boldsymbol{l}, \boldsymbol{\varphi}) b(k - |\boldsymbol{l}|; \boldsymbol{l}, \boldsymbol{\varphi}). \tag{2.11}$$

While the computation of $\mathbb{E}[g(\boldsymbol{L}, \boldsymbol{\Phi})]$ involves the computation of an infinite number of coefficients, in practice only a finite number of coefficients can be computed. In case $\chi^k f(k)$ converges to zero as $k \to \infty$, we can compute $\mathbb{E}[g(\boldsymbol{L}, \boldsymbol{\Phi})]$ up to arbitrary precision by truncating the series after a finite number of terms. We define M to be this number minus one, so that the truncated series consists of $M + 1$ terms. We thus obtain the following computational scheme to evaluate $\mathbb{E}[g(\boldsymbol{L}, \boldsymbol{\Phi})]$:

1. Determine $b(0; \mathbf{0}, \boldsymbol{\varphi})$ by solving the set of equations consisting of all but one of the equations in (2.7) together with (2.8). Compute $f(0)$ according to (2.11), i.e.

$$f(0) = \sum_{\boldsymbol{\varphi} \in \mathscr{S}} g(\mathbf{0}, \boldsymbol{\varphi}) b(0; \mathbf{0}, \boldsymbol{\varphi}). \tag{2.12}$$

2. Let $f(k) := 0$, $k = 1, 2, \ldots$

3. Set $m := 1$.

4. For all $(k; \boldsymbol{l}, \boldsymbol{\varphi}) \in \mathbb{N}^3 \times \mathscr{S}$ with $\boldsymbol{l} \neq \mathbf{0}$ and with $k + |\boldsymbol{l}| = m$, compute $b(k; \boldsymbol{l}, \boldsymbol{\varphi})$ by iteratively solving the equation set (2.6) in increasing order of $(k; \boldsymbol{l}, \boldsymbol{\varphi})$ with respect to $\prec$. Update $f(m)$ according to (2.11).

5. For all $\boldsymbol{\varphi} \in \mathscr{S}$, compute $b(m; \mathbf{0}, \boldsymbol{\varphi})$ by solving the set of equations consisting of all but one of the equations in (2.6) in combination with (2.9). Update $f(m)$ according to (2.11).

6. Set $m := m + 1$. If $m \le M$, return to step 4, otherwise stop. The estimated value for $\mathbb{E}[g(\boldsymbol{L}, \boldsymbol{\Phi})]$ is now given by $\sum_{k=0}^{M} \chi^k f(k)$.

With this computational scheme, performance measures such as the r-th moment of L_i or the cross-moment $\mathbb{E}[L_1 L_2]$ can be computed by taking $g(\boldsymbol{l}, \boldsymbol{\varphi}) = l_i^r$ or $g(\boldsymbol{l}, \boldsymbol{\varphi}) = l_1 l_2$, respectively. Moreover, note that the steady-state probabilities $p(\boldsymbol{n}, \boldsymbol{\psi})$ themselves can be computed through this scheme by taking $g(\boldsymbol{l}, \boldsymbol{\varphi}) = \mathbb{1}_{\{\boldsymbol{l} = \boldsymbol{n}, \boldsymbol{\varphi} = \boldsymbol{\psi}\}}$. We end this section with several remarks.

REMARK 2.3.1. For the numerical evaluation of the performance measures, we compute (2.10) using the corresponding function $g(\boldsymbol{l}, \boldsymbol{\varphi})$ and truncate the power series after the M-th order term. In general, it is hard to say exactly how to choose the value of M in order to achieve a certain degree of accuracy. First, this number depends on the 'degree of symmetry'. If the rates of arrival, service, breakdown and repair do not differ between the first-layer queues and machines, the power series (2.10) generally converges faster than for systems where these rates are queue-dependent or machine-dependent. Secondly, the choice of M also depends heavily on the load offered to the system. For small χ, only a small number of terms has to be computed for the truncated power series to be accurate.

REMARK 2.3.2. It is not guaranteed that the power series (2.5) and (2.10) converge for every value of χ, even if the system is stable. Therefore, it may happen that the power-series algorithm fails for highly asymmetric systems, because (2.5) and (2.10) are divergent. There are two techniques available in the literature to improve the convergence properties of these power series. For an extensive discussion of these methods, see e.g. [42]. The conformal mapping technique attempts to enlarge the radius of convergence by mapping any singularities outside of the circle $|\chi| < \chi^*$. Alternatively, the epsilon algorithm accelerates convergence of a slowly convergent power series or determines a value for a divergent series. This is done by approximating the performance measure under consideration by a sequence of quotients of polynomials.

REMARK 2.3.3. Observe that in Section 2.2, we have assumed the interarrival times, service times, breakdown times and repair times to be exponentially distributed. However, this is not strictly needed to apply the power-series algorithm. In order to use the power-series algorithm, we only need phase-type distributions. For phase-type distributions, the auxiliary vector $\mathbf{\Phi}(t)$ must be expanded to include information on the phase each of the running times is in, in order to preserve the Markov property of the process $\{(\boldsymbol{L}(t), \mathbf{\Phi}(t)), t \geq 0\}$. Therefore, the size of the auxiliary state space $\mathscr{S}$ increases. This may lead to a considerable increase in complexity of the computational scheme, since the equation set (2.6) now contains more equations and more unknowns. For Coxian distributions, however, the increased complexity is limited, since the phases of a Coxian distribution are placed in sequence. Therefore, (2.6) will be a relatively sparse set of equations. Note that up to now, we have made no distinction between the service of type-i products being either resumed or restarted after an interruption, since we assumed the service times to be exponential. However, when assuming phase-type distributed service times, both scenarios can be modelled by choosing the correct auxiliary destination state ψ for the rate $u(\boldsymbol{l}, \boldsymbol{\varphi}, \psi)$ that coincides with the end of a repair of M_i. As the current phase of any service at Q_i is stored in the state $\boldsymbol{\varphi}$, one either takes the state ψ such that it includes the same service phase information in case of service resumption, or such that it refers to the first phase of service in case services are restarted. In the latter case, the current service at Q_i resets to its first phase when M_i becomes operational again.

REMARK 2.3.4. Although we have restricted ourselves thus far to the case of two machines and a single repairman, the power-series algorithm is also applicable for larger numbers of machines and repairmen. For a larger number of machines and first-layer queues, information on the order in which the machines are waiting for repair needs to be included in the auxiliary vector $\mathbf{\Phi}(t)$. Because the dimension of the vector $\boldsymbol{L}(t)$ and the size of the state space $\mathscr{S}$ will increase, the computational complexity increases accordingly. For a larger number of repairmen, no additional non-exponentiality is introduced to the system and thus no additional information needs to be included into $\mathbf{\Phi}(t)$, although the state space $\mathscr{S}$ and the rates $u(\boldsymbol{l}, \boldsymbol{\varphi}, \psi)$ will evidently change.

2.4 Light-traffic behaviour

In Section 2.3, we have derived a computational scheme to numerically compute performance measures. If the power series (2.10) converges, these computations can be performed up to arbitrary precision by truncating the power series and subsequently recursively computing the coefficients $f(k)$. This leads to the question whether the power-series

algorithm can also be used to obtain similar computations in a *symbolic* fashion. In theory, this is possible by running the computational scheme as before, but now using symbolic parameter values instead of numerical values for the rates of arrival, service, breakdown and repair. However, due to constraints in computational resources, only coefficients $f(k)$ up to a small value of k can be computed symbolically before the computations become too cumbersome. The set of equations (2.6) becomes increasingly hard to solve, as the expressions for the terms $b(k;\boldsymbol{l},\boldsymbol{\varphi})$ quickly become very large as k increases.

The number of coefficients that can be computed symbolically in practice is generally not enough to obtain an accurate approximation for general values of χ. However, as χ becomes smaller, the higher-order terms become increasingly negligible. Therefore, the so-called light-traffic behaviour of a performance measure as χ tends to zero can be identified symbolically. We do so for the performance measures $\mathbb{E}[L_1]$ and $\mathbb{E}[L_1L_2]$ in Sections 2.4.1 and 2.4.2, respectively. For the sake of clarity, we will refer to the k-th order coefficient $f(k)$ in (2.10) corresponding to $g(\boldsymbol{l},\boldsymbol{\varphi}) = l_1$ as $f_1(k)$ in the sequel. Similarly, $f_2(k)$ denotes the k-th order coefficient corresponding to $g(\boldsymbol{l},\boldsymbol{\varphi}) = l_1l_2$.

2.4.1 Marginal queue length

We are interested in the light-traffic behaviour of the marginal queue length L_1 in the variable $\chi = \hat{\rho}_1$. More specifically, we consider the behaviour of the mean of L_1 as a function of the relative load $\hat{\rho}_1$ as $\hat{\rho}_1$ goes to zero. By taking $g(\boldsymbol{l},\boldsymbol{\varphi}) = l_1$ and running the power-series algorithm with $M = 2$, we obtain the following expression for $\mathbb{E}[g(\boldsymbol{L},\boldsymbol{\Phi})] = \mathbb{E}[L_1]$:

$$\mathbb{E}[L_1] = f_1(0) + f_1(1)\hat{\rho}_1 + f_1(2)\hat{\rho}_1^2 + \mathcal{O}(\hat{\rho}_1^3), \tag{2.13}$$

where $\mathcal{O}(\hat{\rho}_1^3)$ represents third-order and higher-order terms in $\hat{\rho}_1$. Furthermore, we have that $f_1(0) = 0$, since $g(\boldsymbol{0},\boldsymbol{\varphi}) = 0$ in (2.12). This is explained by the fact that there are no type-1 arrivals for $\hat{\rho}_1 = 0$, and thus there never is any product in Q_1. The coefficient $f_1(1)$ equals $\frac{d}{d\hat{\rho}_1}\mathbb{E}[L_1]|_{\hat{\rho}_1=0}$, the derivative of the mean of L_1 with respect to $\hat{\rho}_1$ evaluated at $\hat{\rho}_1 = 0$. Computing $f_1(1)$ leads to a closed-form expression in the service rate of M_1 as well as the breakdown and repair rates of each of the machines. Since this term is too large to display in its entirety, we give the expressions for $\frac{d}{d\hat{\rho}_1}\mathbb{E}[L_1]|_{\hat{\rho}_1=0}$ in each of the model parameters separately in Table 2.1. When giving the derivative in each of these parameters, we assume all other parameters to be equal to one. From these results, we see that $\frac{d}{d\hat{\rho}_1}\mathbb{E}[L_1]|_{\hat{\rho}_1=0}$ is increasing in μ_1 and decreasing in ν_1 and ν_2. The latter is not surprising, as it intuitively makes sense that the queue length generally decreases (in some sense) as the repair rates increase. Moreover, we note that the denominators of the terms in the expressions only involve the model parameters in the form of polynomials of at most the second order.

It is important to observe that the expression $f_1(1)$ also represents the first-order derivative of higher moments of L_1 as $\hat{\rho}_1$ goes to zero. In other words, the first-order derivative of $\mathbb{E}[L_i^r]$ with respect to $\hat{\rho}_1$ evaluated at $\hat{\rho}_1 = 0$ is independent of r. This can be explained by careful inspection of $f(1)$ in (2.11). The first-order term $f_1(1)$ only involves values of $g(\boldsymbol{l},\boldsymbol{\varphi})$ for which $|\boldsymbol{l}| \in \{0,1\}$, which implies that l_1 can also only take the values zero and one. To inspect $\mathbb{E}[L_i^r]$, we take $g(\boldsymbol{l},\boldsymbol{\varphi}) = l_1^r$. Since $l_1 \in \{0,1\}$, the function $g(\boldsymbol{l},\boldsymbol{\varphi})$ can only evaluate to the values $0^r = 0$ or $1^r = 1$ irrespective of $r > 0$.

The application of the power-series algorithm in a symbolic manner also allows us to find closed-form expressions for the second-order derivative $\frac{d^2}{d\hat{\rho}_1^2}\mathbb{E}[L_1]|_{\hat{\rho}_1=0} = 2f_1(2)$. In

TABLE 2.1: Expressions for $\frac{d}{d\hat{\rho}_1}\mathbb{E}[L_1]|_{\hat{\rho}_1=0}$ in each of the model parameters.

Model parameter	$\frac{d}{d\hat{\rho}_1}\mathbb{E}[L_1]\|_{\hat{\rho}_1=0} = f_1(1)$
μ_1	$\frac{26}{25} + \frac{8\mu_1}{25} - \frac{3}{25(3+\mu_1)}$
σ_1	$1 + \frac{9}{49(3+\sigma_1)} - \frac{36}{7(2+3\sigma_1)^2} + \frac{120}{49(2+3\sigma_1)}$
σ_2	$\frac{4}{3} - \frac{3}{49(3+\sigma_2)} - \frac{13}{21(2+3\sigma_2)^2} + \frac{9}{49(2+3\sigma_2)}$
ν_1	$\frac{75}{64} + \frac{135}{256(1+2\nu_1)^2} + \frac{21}{256(1+2\nu_1)} + \frac{567}{256(3+2\nu_1)^2} - \frac{21}{256(3+2\nu_1)}$
ν_2	$\frac{5}{4} - \frac{13}{75(3+\nu_2)} + \frac{27}{20(1+2\nu_2)^2} - \frac{57}{100(1+2\nu_2)} - \frac{1}{2(3+2\nu_2)^2} + \frac{11}{12(3+2\nu_2)}$

TABLE 2.2: Expressions for $\frac{d^2}{d\hat{\rho}_1^2}\mathbb{E}[L_1]|_{\hat{\rho}_1=0}$ in each of the model parameters.

Model parameter	$\frac{d^2}{d\hat{\rho}_1^2}\mathbb{E}[L_1]\|_{\hat{\rho}_1=0} = 2f_1(2)$
μ_1	$\frac{226}{125} + \frac{88\mu_1}{125} - \frac{108}{125(3+\mu_1)^3} - \frac{18}{125(3+\mu_1)^2} + \frac{18}{25(3+\mu_1)}$
σ_1	$2 - \frac{36}{343(3+\sigma_1)^3} + \frac{222}{2401(3+\sigma_1)^2} + \frac{4538}{16807(3+\sigma_1)} + \frac{272}{343(2+3\sigma_1)^3}$ $- \frac{26800}{2401(2+3\sigma_1)^2} + \frac{87228}{16807(2+3\sigma_1)}$
σ_2	$\frac{8}{3} - \frac{12}{343(3+\sigma_2)^3} - \frac{500}{2401(3+\sigma_2)^2} - \frac{3784}{16807(3+\sigma_2)} - \frac{68}{343(2+3\sigma_2)^3}$ $- \frac{10952}{7203(2+3\sigma_2)^2} + \frac{11352}{16807(2+3\sigma_2)}$
ν_1	$\frac{2385}{1024} - \frac{459}{8192(1+2\nu_1)^3} + \frac{22725}{16384(1+2\nu_1)^2} - \frac{11673}{16384(1+2\nu_1)}$ $+ \frac{19683}{8192(3+2\nu_1)^3} + \frac{105309}{16384(3+2\nu_1)^2} + \frac{12249}{16384(3+2\nu_1)}$
ν_2	$\frac{5}{2} - \frac{52}{1125(3+\nu_2)^3} - \frac{2312}{5625(3+\nu_2)^2} - \frac{48194}{84375(3+\nu_2)} - \frac{1107}{4000(1+2\nu_2)^3}$ $+ \frac{110367}{40000(1+2\nu_2)^2} - \frac{106017}{100000(1+2\nu_2)} - \frac{283}{288(3+2\nu_2)^3} - \frac{59}{64(3+2\nu_2)^2}$ $+ \frac{1903}{864(3+2\nu_2)}$

Table 2.2, we give this expression in each of the model parameters. Again, we assume the other parameters to be equal to one. As before, we see that $\frac{d^2}{d\hat{\rho}_1^2}\mathbb{E}[L_1]|_{\hat{\rho}_1=0}$ is increasing in μ_1 and decreasing in ν_1 and ν_2. Furthermore, note that the denominators of the expressions only involve the model parameters in a polynomial fashion up to order three. This is not surprising, as the expressions for the first derivative only involve the parameters up to a second order.

REMARK 2.4.1. If we wish to compute the light-traffic behaviour of the moments of L_2, we perform similar computations to the above, or we simply renumber the queues.

REMARK 2.4.2. Note that the computation of $f_1(1) = \frac{d}{d\hat{\rho}_1}\mathbb{E}[L_1]|_{\hat{\rho}_1=0}$ is also possible using Little's law:

$$\mathbb{E}[S_1]|_{\hat{\rho}_1=0} = \frac{\mathbb{E}[L_1]|_{\hat{\rho}_1=0}}{\lambda_1} = \frac{f_1(1)\hat{\rho}_1 + \mathcal{O}(\hat{\rho}_1^2)}{\lambda_1}\bigg|_{\hat{\rho}_1=0} = \frac{f_1(1)}{\mu_1 m_{C,1}} = \frac{\frac{d}{d\hat{\rho}_1}\mathbb{E}[L_1]|_{\hat{\rho}_1=0}}{\mu_1 m_{C,1}}, \quad (2.14)$$

where $\mathbb{E}[S_1]|_{\hat{\rho}_1=0}$ is the mean sojourn time of a type-1 product, conditioned on the event there are no other products in the system. This sojourn time consists of the actual service

TABLE 2.3: Expressions for $\frac{d^2}{d\hat{\rho}_1^2}\mathbb{E}[L_1L_2]|_{\hat{\rho}_1=0}$ in each of the model parameters.

Model parameter	$\frac{d^2}{d\hat{\rho}_1^2}\mathbb{E}[L_1L_2]\|_{\hat{\rho}_1=0} = 2f_2(2)$
μ_1	$-\frac{1816d}{3375} + \frac{3646d\mu_1}{1125} + \frac{413d\mu_1^2}{375} + \frac{36d}{125(3+\mu_1)} + \frac{32(373d+143d\mu_1)}{3375\left(8+13\mu_1+3\mu_1^2\right)}$
μ_2	$\frac{413d}{375} + \frac{133d}{50\mu_2} + \frac{4d}{125(3+\mu_2)} + \frac{4357d+1235d\mu_2}{750\left(8+13\mu_2+3\mu_2^2\right)}$
σ_1	$-\frac{20d}{1+\sigma_1} + \frac{96d}{343(3+\sigma_1)^2} - \frac{3560d}{2401(3+\sigma_1)} - \frac{120d}{2197(5+\sigma_1)} - \frac{110360d}{1911(2+3\sigma_1)^3}$ $-\frac{5295776d}{173901(2+3\sigma_1)^2} + \frac{425228092d}{5274997(2+3\sigma_1)}$
σ_2	$\frac{97d}{27} + \frac{4d\sigma_2}{3} + \frac{96d}{343(3+\sigma_2)^2} - \frac{4232d}{2401(3+\sigma_2)} - \frac{480d}{2197(5+\sigma_2)}$ $-\frac{27590d}{17199(2+3\sigma_2)^3} - \frac{235012d}{57967(2+3\sigma_2)^2} - \frac{5574427d}{5274997(2+3\sigma_2)}$
ν_1	$\frac{4779d}{896} - \frac{364d}{125(3+\nu_1)} + \frac{3267d}{5120(1+2\nu_1)^3} + \frac{357131d}{51200(1+2\nu_1)^2}$ $-\frac{3557887d}{6912000(1+2\nu_1)} - \frac{189855d}{13312(3+2\nu_1)^3} + \frac{5779335d}{346112(3+2\nu_1)^2}$ $-\frac{78477979d}{4499456(3+2\nu_1)} + \frac{17756000d}{415233(17+7\nu_1)}$
ν_2	$\frac{531d}{224} + \frac{34d}{1+\nu_2} - \frac{364d}{1125(3+\nu_2)} + \frac{3267d}{1280(1+2\nu_2)^3} + \frac{313571d}{12800(1+2\nu_2)^2}$ $-\frac{60000667d}{1728000(1+2\nu_2)} - \frac{21095d}{3328(3+2\nu_2)^3} - \frac{4655195d}{259584(3+2\nu_2)^2}$ $-\frac{291320479d}{10123776(3+2\nu_2)} + \frac{710240d}{415233(17+7\nu_2)}$

requirement, the time the product needs to wait before M_1 takes the product into service and the downtime M_1 suffers during the service of the product. The mean of the first term obviously equals μ_1^{-1}. The means of the latter two terms can be computed by studying the continuous-time Markov chain $\{\mathbf{\Phi}(t), t \geq 0\}$. Eventually, this leads to an expression for $\mathbb{E}[S_1]|_{\hat{\rho}_1=0}$, which in turn leads to an expression for $\frac{d}{d\hat{\rho}_1}\mathbb{E}[L_1]|_{\hat{\rho}_1=0}$ due to (2.14).

2.4.2 Joint queue length

In this section, we discuss the light-traffic behaviour of $\mathbb{E}[L_1L_2]$, the cross-moment of the queue lengths in the extended machine repair model, as a function of $\hat{\rho}_1$. We study instances of the model for which both of the arrival rates tend to zero while we preserve the relative values; i.e. we assume that $\lambda_2 = d\lambda_1$ at all times for a constant $d > 0$. This means that we set $a^{(1)}(\boldsymbol{l}, \boldsymbol{\varphi}) = \mu_1 m_{C,1}$ and $a^{(2)}(\boldsymbol{l}, \boldsymbol{\varphi}) = d\mu_1 m_{C,1}$ while we let λ_1 (or $\hat{\rho}_1$) go to zero. Furthermore, we take $g(\boldsymbol{l}, \boldsymbol{\varphi}) = l_1 l_2$. By running the computational scheme as given in Section 2.3.2 with $M = 2$, we obtain the following expression for $\mathbb{E}[L_1L_2]$:

$$\mathbb{E}[L_1L_2] = f_2(0) + f_2(1)\hat{\rho}_1 + f_2(2)\hat{\rho}_1^2 + \mathcal{O}(\hat{\rho}_1^3). \tag{2.15}$$

Like before, we have that $f_2(0) = 0$, because $g(\mathbf{0}, \boldsymbol{\varphi}) = 0$ for all $\boldsymbol{\varphi} \in \mathcal{S}$ in (2.12). We also have that $f_2(1) = 0$ due to (2.11). The coefficient $f_2(1)$ only involves values of $g(\boldsymbol{l}, \boldsymbol{\varphi})$ for which $0 \leq l_1 + l_2 \leq 1$. Within this domain, there is no combination (l_1, l_2) for which $l_1 l_2 > 0$. Therefore, the most prominent light-traffic behaviour is captured by the term $f_2(2)$.

Going back to the derivatives of the cross-moment, we have that the first-order derivative of $\mathbb{E}[L_1L_2]$ vanishes at $\hat{\rho}_1 = 0$, since $f_2(1) = 0$. By (2.15), we have for the

second-order derivative that $\frac{d^2}{d\hat{\rho}_1^2}\mathbb{E}[L_1L_2]|_{\hat{\rho}_1=0} = 2f_2(2)$. By evaluation of the computational scheme up to $M = 2$, we obtain a closed-form expression for this second-order derivative evaluated at $\chi = \hat{\rho}_1 = 0$. Again, we give the expression separately in each of the model parameters in Table 2.3 while assuming each of the others to be equal to one. As in the previous case, we note that the numerators and the denominators of the terms in $\frac{d^2}{d\hat{\rho}_1^2}\mathbb{E}[L_1L_2]|_{\hat{\rho}_1=0}$ only involve the model parameters in a polynomial fashion up to order three. For the service rates μ_1 and μ_2, the expressions are equivalent. If we let λ_2 scale along with λ_1 such that $\rho_1 = \rho_2$ and $d = \frac{\mu_2}{\mu_1}$, we even have that the expressions in Table 2.3 pertaining to μ_1 and μ_2 are the same. The parameter $\hat{\rho}_1$ thus depends on the service rates μ_1 and μ_2 in the same way. Also the corresponding equations for the breakdown rates σ_1 and σ_2, as well as those for the repair rates ν_1 and ν_2, are equivalent. This is not surprising, because $\mathbb{E}[L_1L_2]$ behaves symmetrically with respect to both of the queue lengths and is therefore equally sensitive to characteristics of either of the machines.

3

HEAVY-TRAFFIC ASYMPTOTICS

Having studied the light-traffic asymptotics of the extended machine repair model in the previous chapter, the question arises whether any results for its heavy-traffic asymptotics can be obtained, i.e. the behaviour of the performance measures when the arrival rates of products are scaled to such a proportion that the first-layer queues are on the verge of instability. In this chapter, we derive heavy-traffic asymptotics for a very generic model that subsumes the extended machine repair model. We study a network of parallel single-server queues where the speeds of the servers may vary over time and are governed by a single continuous-time Markov chain. We obtain heavy-traffic limits for the distributions of the joint workload, waiting-time and queue length processes. We do so by using a functional central limit theorem approach, which requires the interchange of steady-state and heavy-traffic limits. The marginals of these limiting distributions are shown to be exponential with rates that can be computed by matrix-analytic methods. Moreover, we show how to numerically compute the joint distributions by viewing the limit processes as multi-dimensional semi-martingale reflected Brownian motions in the non-negative orthant. We also demonstrate how to use these results for the performance evaluation of the extended machine repair model. As is the case with the light-traffic results in Chapter 2, the heavy-traffic insights that we gain in this chapter will serve as a building block for the approximations that we derive in Chapter 4.

3.1 Introduction

In this chapter, we study a parallel network of N single-server queues, which can be regarded as a generalisation of the extended machine repair model. The speeds of the servers vary over time and are mutually dependent. More specifically, we assume that these service speeds are governed by a single irreducible, continuous-time Markov chain with a finite state space. For this network, we are interested in both the marginal and the joint workload processes for each of the queues, as well as the processes describing the virtual waiting time and the queue length. Stationary distributions for these processes are difficult to obtain, since the workload process pertaining to one queue, as well as the virtual waiting-time process and the queue length process pertaining to this queue, is correlated with the corresponding processes of the other queues. Our goal in this chapter

is to derive the heavy-traffic behaviour of the network by obtaining the limiting stationary distributions of the aforementioned processes.

Apart from our intended analysis of the extended machine repair model, the study of this general network is motivated by the fact that multi-queue performance models with time-varying and mutually dependent service speeds find a wide variety of other applications. An example is the field of *wireless networks*, where multiple users transmit data packets through a wireless medium at speeds that are typically varying over time and mutually dependent, e.g. due to phenomena such as 'shadow fading' (cf. [244]). Another such application constitutes an *I/O subsystem* of an application server (see e.g. [251]), in which the content of multiple I/O buffers is transferred to clients at varying and mutually dependent speeds due to the varying level of congestion of the application server's network connection. A final example is given by the phenomenon of *garbage collection* in multi-threaded computer systems (cf. [225]). Typically, when the total memory utilisation in such a system exceeds a certain threshold, the processing speeds of the threads are temporarily reduced, and are as a result mutually dependent.

Queueing models with service speeds that vary over time have received attention in multiple settings in the literature. In practice, service speeds may be dependent on factors such as the workload present in the system, which leads to the formulation of queues with state-dependent service rates; see e.g. [31] for an overview. Another branch of work on time-varying service speeds is that of service rate control, where the aim is to minimise waiting and capacity costs (e.g. [20, 105, 230, 270]) or to optimise a trade-off between service quality and service speed (e.g. [127]) based on the state of the system by dynamically varying the service speed. In our case, the service speeds depend on an external environment that is governed by a continuous-time Markov chain. Analyses of single-server queueing models with Markov-modulated service speeds can be found in [115, 173, 182, 201, 234]. However, none of these papers concern themselves with the derivation of heavy-traffic asymptotics. In this chapter, we focus on a queueing network where the service speeds of *all* servers in the network are simultaneously governed by a *single* continuous-time Markov chain. This allows us to incorporate mutual dependencies between the service speeds into the model. Conceptually, there are no additional challenges in obtaining heavy-traffic results for the queueing network with multiple queues compared to the single-queue case, although deriving the results for the multi-queue case is more cumbersome at times.

We are mainly interested in the heavy-traffic asymptotics of the network of queues. The study of queues in heavy traffic was initiated by Kingman with a series of papers in the 1960s, starting with [140]; see [141] for an overview of these early results. These papers were largely focused on the use of Laplace transforms. In our case, however, Laplace transforms for the stationary distribution of the total workload process or even the workload process for a queue in isolation are hard to obtain. The workload process of a queue in isolation can in principle be modelled as a reflected Markov additive process. For the definition and an overview of the standard theory on Markov additive processes, see [19, Section XI.2]. However, the stationary distribution of the workload process is not easily derived from that. For example, standard techniques such as relating the Laplace transforms of the stationary workload conditional on the states of the modulator to each other typically lead to a linear system with a number of equations smaller than the number of unknowns, defying straightforward solutions, as shown in [129]. Less straightforward computations might involve studying the singularities of the characterising matrix expo-

nent pertaining to the reflected Markov additive process (cf. [129]). In the past, stationary distributions for special cases of reflected Markov additive processes have also been analysed by studying their spectral expansion (e.g. [177]) or by determining the boundary probabilities in terms of the solution of a generalised eigenvalue problem (e.g. [245]).

As it is not clear that the approach via Laplace transforms will work in our case, we will use a functional central limit theorem approach mainly developed by Iglehart and Whitt; see [275] for an overview. This is not always trivial; see for example [78, 149]. Heavy-traffic approximations for generalised Jackson networks were studied in [56, 104]. However, the model that we consider does not fall in the framework of generalised Jackson networks. Instead, we tailor more classical arguments for single-node systems to our setting. An advantage of our approach is that it can be extended to allow for variations or generalisations of our model. For example, it is assumed that the workload input processes of the queues are compound Poisson processes. As we will see in the sequel, however, our approach for deriving heavy-traffic asymptotics still remains valid under relaxed assumptions if Lemma 3.3.2 can be proved for this more general setting.

As we study networks with general service speeds, the generic model also covers a class of queues with service interruptions. Heavy-traffic asymptotics for single-server queues with vacations have been studied in [136]. Related but different problems are networks with interruptions of which durations and frequency scale with the traffic intensity, and have been studied in [59, 136] and [275, Section 14.7]. As opposed to these models, our model allows the durations of consecutive service interruptions, which we assume to be independent of the traffic intensity, to be interdependent through the Markovian random environment (see also [62]), and the interruptions are not restricted to a point in time the queue empties.

For the network studied in this chapter, we find that the marginal workload, virtual waiting-time and queue length processes pertaining to a queue in isolation exhibit state-space collapse under heavy-traffic assumptions and have exponential limiting distributions. Moreover, we show that the limiting distribution of the joint workload process, as well as that of the joint virtual waiting-time process and the joint queue length process, corresponds to the stationary distribution of an N-dimensional semi-martingale reflected Brownian motion with state space $\mathbb{R}_+^N$ (see e.g. [60, Theorem 6.2] for a definition). The reflection matrix corresponding to this semi-martingale reflected Brownian motion is an identity matrix, so that positive conclusions about the existence of a stationary distribution can be drawn (cf. [119]). However, computing this distribution is challenging. The conditions needed for the stationary distribution to have a product form do not apply to our model, and results such as those of [82] seem hard to translate to our setting. In this chapter, we therefore show how to use the numerical methods developed in [70] for steady-state analysis of multi-dimensional semi-martingale reflected Brownian motions to analyse the joint limiting distribution of the stationary workload process. This allows us to compute quantities such as the correlation coefficients between the marginal components.

The rest of this chapter is organised as follows. Section 3.2 describes the generic network in more detail, gives the necessary notation and gives several preliminary results. In Section 3.3, we derive the heavy-traffic limit for a properly scaled workload process pertaining to this network, and observe that the stationary distribution of the marginal workload processes converges to an exponential distribution. Section 3.4 extends these results to heavy-traffic limits for the virtual waiting-time and queue length processes. Fi-

nally, in Section 3.5, we study how one can compute the joint distribution of the limiting processes pertaining to the workloads, virtual waiting times and the queue lengths by viewing these as semi-martingale reflected Brownian motions. We also show how to apply these results to the extended machine repair model. From the resulting numerical computations, we conclude that even in a heavy-traffic regime, the interaction between the layers and the correlations between the first-layer queues can be significant. By means of simulation results, we also show that the obtained heavy-traffic results give rise to accurate approximations for considerably loaded systems, which marks the usefulness of the heavy-traffic analysis that we perform from an application perspective.

3.2 Notation and preliminaries

In this section, we introduce the generic model that we study in this chapter as well as its notation, and we present several preliminary results.

We study a network consisting of N parallel single-server queues $Q_1, \ldots, Q_N$, each with its own dedicated arrival stream. Type-i customers arrive at Q_i according to a Poisson process with rate λ_i and have a service requirement distributed according to a random variable B_i with finite first two moments $\mathbb{E}[B_i]$ and $\mathbb{E}[B_i^2]$. In particular, we represent by $B_{i,j}$ the service requirement of the j-th arriving type-i customer. We assume the service requirements of all customers to be mutually independent. Further, we denote by $\{N_i(t), t > 0\}$ a unit-rate Poisson process. Then, the cumulative workload that enters Q_i during the time interval $[0, t)$ is given by

$$V_i(\lambda_i t) = \sum_{j=1}^{N_i(\lambda_i t)} B_{i,j},$$

where the arrival rate is left as part of the argument, as this will prove to be useful for heavy-traffic scaling purposes in the sequel. In the remainder of this chapter, we will refer to $\{V_i(t), t \geq 0\}$ as the arrival process of Q_i. The mean corresponding to this arrival process is given by $m_{V,i} = \mathbb{E}[V_i(1)] = \mathbb{E}[B_i]$. Similarly, the variance is given by $\sigma^2_{V,i} = \mathrm{Var}[V_i(1)] = \mathbb{E}[N_i(1)]\mathrm{Var}[B_i] + \mathrm{Var}[N_i(1)]\mathbb{E}[B_i]^2 = \mathrm{Var}[B_i] + \mathbb{E}[B_i]^2 = \mathbb{E}[B_i^2]$. Note that the arrival process has stationary and independent increments, so that $t^{-1}\mathbb{E}[V_i(t)] = m_{V,i}$ and $t^{-1}\mathrm{Var}[V_i(t)] = \sigma^2_{V,i}$ for any $t > 0$.

The service speeds of the N servers serving $Q_1, \ldots, Q_N$ may vary over time and are mutually dependent. More specifically, the joint process of these service speeds is modulated by a single irreducible, stationary, continuous-time Markov chain $\{\Phi(t), t \geq 0\}$ with finite state space $\mathscr{S}$ and invariant probability measure $\boldsymbol{\pi} = (\pi_i)_{i \in \mathscr{S}}$. When this Markov chain resides in the state $\omega \in \mathscr{S}$, the server of Q_i drains its queue at service rate $\phi_i(\omega)$. As a consequence, we have that the workload that the server of Q_i has been capable of processing during the time interval $[0, t)$ is represented by

$$C_i(t) = \int_{s=0}^{t} \phi_i(\Phi(s))ds.$$

We will also refer to the process $\{C_i(t), t \geq 0\}$ as the cumulative service process of Q_i. Note that, as the continuous-time Markov chain $\{\Phi(t), t \geq 0\}$ is in stationarity, the increments

of the process $\{C_i(t), t \geq 0\}$ are also stationary. The mean corresponding to the process $\{C_i(t), t \geq 0\}$ is given by

$$m_{C,i} = \mathbb{E}[C_i(1)] = \int_{s=0}^{1} \sum_{\omega \in \mathcal{S}} \phi_i(\omega) \mathbb{P}(\Phi(s) = \omega) ds = \sum_{\omega \in \mathcal{S}} \phi_i(\omega) \pi_\omega.$$

Since the C_i-process has stationary increments, it holds that $t^{-1}\mathbb{E}[C_i(t)] = m_{C,i}$ for any $t > 0$. We denote the asymptotic variance $\lim_{t\to\infty} t^{-1}\text{Var}[C_i(t)]$ by $\sigma^2_{C,i}$. Similarly, the long-run time-averaged covariance between the cumulative service processes of the servers at Q_i and Q_j is represented by $\gamma^C_{i,j} = \lim_{t\to\infty} \frac{1}{t}\text{Cov}[C_i(t), C_j(t)]$. Computing expressions for $\sigma^2_{C,i}$ and $\gamma^C_{i,j}$ is not trivial. We focus on this problem in Section 3.5.1.

A queue Q_i is said to be stable if the expected amount of arriving work $\lambda_i \mathbb{E}[B_i]$ per time unit is smaller than the average workload $m_{C,i}$ that its server is capable of processing per time unit. Equivalently, Q_i is stable if its load, defined as $\rho_i = \frac{\lambda_i \mathbb{E}[B_i]}{m_{C,i}}$, is less than one. We are interested in the performance of the network of queues in heavy traffic, i.e. the case for which the arrival rates $\lambda_1, \ldots, \lambda_N$ are scaled so that $(\rho_1, \ldots, \rho_N) \to \mathbf{1}$. For this purpose, it is convenient to introduce the index r. In the r-th system, each arrival rate λ_i is taken so that $\beta_i(1-\rho_i)^{-1} = r$, where the β_i parameters control the rate at which the arrival rates are scaled by r, while the series of service requirements $B_{i,1}, B_{i,2}, \ldots$ and the C_i-processes are not scaled by r. The heavy-traffic limit for any performance measure of the system corresponds to the limit $r \to \infty$. We denote by $\lambda_{i,r}$ the arrival rate of type-i customers corresponding to the r-th system, so that $\lambda_{i,r} \to \frac{m_{C,i}}{\mathbb{E}[B_i]}$ when $r \to \infty$. For notational convenience, we write for two functions $f(r)$ and $g(r)$ that $f(r) = o(g(r))$ if $\lim_{r\to\infty} f(r)/g(r) = 0$.

For purposes that will become clear in the sequel, we now state heavy-traffic limits for the primitive processes that are scaled in time by a factor r^2. First, for the scaled arrival processes, we observe that $\mathbb{E}[V_i(\lambda_{i,r} r^2 t)] = \lambda_{i,r} r^2 \mathbb{E}[B_i] t$. As the arrival processes constitute independent renewal reward processes, the functional central limit theorem for renewal reward processes (see e.g. [275, Theorem 7.4.1]) implies that

$$\left\{ \left(\frac{V_1(\lambda_{1,r} r^2 t) - \lambda_{1,r} r^2 \mathbb{E}[B_1] t}{\sqrt{\lambda_{1,r}} r}, \ldots, \frac{V_N(\lambda_{N,r} r^2 t) - \lambda_{N,r} r^2 \mathbb{E}[B_N] t}{\sqrt{\lambda_{N,r}} r} \right), t \geq 0 \right\} \xrightarrow{d} \{\boldsymbol{Z}_V(t), t \geq 0\} \quad (3.1)$$

as $r \to \infty$, where $\{\boldsymbol{Z}_V(t), t \geq 0\}$ is an N-dimensional Brownian motion with zero drift and covariance matrix $\boldsymbol{\Gamma}^V = diag(\sigma^2_{V,1}, \ldots, \sigma^2_{V,N})$.

Similarly, after observing that $\mathbb{E}[C_i(r^2 t)] = m_{C,i} r^2 t$, it follows from results in [274] that the time-scaled cumulative service processes satisfy

$$\left\{ \left(\frac{C_1(r^2 t) - m_{C,1} r^2 t}{r}, \ldots, \frac{C_n(r^2 t) - m_{C,N} r^2 t}{r} \right), t \geq 0 \right\} \xrightarrow{d} \{\boldsymbol{Z}_C(t), t \geq 0\} \quad (3.2)$$

as $r \to \infty$, where $\{\boldsymbol{Z}_C(t), t \geq 0\}$ is an N-dimensional Brownian motion with zero drift and covariance matrix $\boldsymbol{\Gamma}^C$ with elements $\Gamma^C_{i,j} = \gamma^C_{i,j}$. Alternatively, this result follows from the functional central limit theorem for Markov additive processes obtained in [229, Theorem 3.4]. Using the results of [229], we will show how to obtain expressions for $\gamma^C_{i,j}$ in Section 3.5.1.

A heavy-traffic limit for the joint scaled net-input process now follows by combining (3.1) and (3.2) with the observation that $\frac{\lambda_{i,r} r^2 \mathbb{E}[B_i] t - m_{C,i} r^2 t}{r} = -\beta_i m_{C,i} t$. In particular, this leads to

$$\left\{ \left(\frac{V_1(\lambda_{1,r} r^2 t) - C_1(r^2 t)}{r}, \ldots, \frac{V_N(\lambda_{N,r} r^2 t) - C_N(r^2 t)}{r} \right), t \geq 0 \right\} \xrightarrow{d} \{ \boldsymbol{Z}(t), t \geq 0 \} \quad (3.3)$$

as $r \to \infty$, where $\{\boldsymbol{Z}(t) = (Z_1(t), \ldots, Z_N(t)), t \geq 0\}$ is an N-dimensional Brownian motion with drift vector $\boldsymbol{\mu} = (-\beta_1 m_{C,1}, \ldots, -\beta_N m_{C,N})$ and covariance matrix

$$\boldsymbol{\Gamma} = diag\left(\frac{m_{C,1}}{\mathbb{E}[B_1]} \sigma^2_{V,1}, \ldots, \frac{m_{C,N}}{\mathbb{E}[B_N]} \sigma^2_{V,N} \right) + \boldsymbol{\Gamma}^C. \quad (3.4)$$

We now derive a representation of the amount of work present in each of the queues. Let $\{\boldsymbol{W}_r(t) = (W_{1,r}(t), \ldots, W_{N,r}(t)), t \geq 0\}$ be the process that describes the workload in each queue of the r-th system at time t and let $\boldsymbol{W}_r = (W_{1,r}, \ldots, W_{N,r}) = \boldsymbol{W}_r(\infty)$ denote the workload in the system in steady state. The processes $\{\boldsymbol{D}_r(t), t \geq 0\}$ and $\{\boldsymbol{L}_r(t), t \geq 0\}$, as well as $\boldsymbol{D}_r$ and $\boldsymbol{L}_r$, are similarly defined for the virtual waiting time (the delay faced by an imaginary customer arriving at time t) and the queue length (excluding the customer in service), respectively.

The workload $W_{i,r}(t)$ present in Q_i at time t can be represented by the one-sided reflection of the net-input process $\{V_i(\lambda_{i,r} t) - C_i(t), t \geq 0\}$ under the assumption that $W_{i,r}(0) = 0$:

$$\begin{aligned} W_{i,r}(t) &= V_i(\lambda_{i,r} t) - C_i(t) - \inf_{s \in [0,t]} \{V_i(\lambda_{i,r} s) - C_i(s)\} \\ &= \sup_{s \in [0,t]} \{V_i(\lambda_{i,r} t) - V_i(\lambda_{i,r} s) - (C_i(t) - C_i(s))\}. \end{aligned} \quad (3.5)$$

As the joint cumulative service process $\{(C_1(t), \ldots, C_N(t)), t \geq 0\}$ has stationary increments, it holds that

$$(C_1(t) - C_1(s), \ldots, C_N(t) - C_N(s)) \stackrel{d}{=} (C_1(t-s), \ldots, C_N(t-s)).$$

Furthermore, since the arrival processes are independent and since compound Poisson processes have time-reversible increments, we also have that

$$\begin{aligned} &\big(V_1(\lambda_{1,r} t) - V_1(\lambda_{1,r} s), \ldots, V_N(\lambda_{N,r} t) - V_N(\lambda_{N,r} s)\big) \\ &\stackrel{d}{=} \big(V_1(\lambda_{1,r}(t-s)), \ldots, V_N(\lambda_{N,r}(t-s))\big). \end{aligned}$$

Due to this, we have by (3.5) that $\boldsymbol{W}_r(t)$ satisfies

$$\begin{aligned} \boldsymbol{W}_r(t) &\stackrel{d}{=} \left(\sup_{s \in [0,t]} \{V_1(\lambda_{1,r}(t-s)) - C_1(t-s)\}, \ldots, \sup_{s \in [0,t]} \{V_N(\lambda_{N,r}(t-s)) - C_N(t-s)\} \right) \\ &= \left(\sup_{s \in [0,t]} \{V_1(\lambda_{1,r}(s)) - C_1(s)\}, \ldots, \sup_{s \in [0,t]} \{V_N(\lambda_{N,r}(s)) - C_N(s)\} \right). \end{aligned}$$

By letting $t \to \infty$, this results in

$$\boldsymbol{W}_r \stackrel{d}{=} \left(\sup_{s \geq 0} \{V_1(\lambda_{1,r} s) - C_1(s)\}, \ldots, \sup_{s \geq 0} \{V_N(\lambda_{N,r} s) - C_N(s)\} \right). \quad (3.6)$$

In this study, we are particularly interested in the distribution of the scaled workload $\overline{\boldsymbol{W}}_r = \frac{\boldsymbol{W}_r}{r}$ (as well as the similarly defined scaled virtual waiting time $\overline{\boldsymbol{D}}_r$ and scaled queue length $\overline{\boldsymbol{L}}_r$) in heavy traffic, i.e. as $r \to \infty$. It is easily seen from (3.6) that the scaled workload can be written in terms of the similarly scaled net-input process. That is, after scaling time by a factor r^2, we have

$$\overline{\boldsymbol{W}}_r \stackrel{d}{=} \left(\sup_{t\geq 0} \left\{ \frac{V_1(\lambda_{1,r} r^2 t) - C_1(r^2 t)}{r} \right\}, \ldots, \sup_{t\geq 0} \left\{ \frac{V_N(\lambda_{N,r} r^2 t) - C_N(r^2 t)}{r} \right\} \right). \tag{3.7}$$

3.3 Heavy-traffic asymptotics of the workload

In this section, we derive the following heavy-traffic asymptotic result for the scaled workload $\overline{\boldsymbol{W}}_r$.

THEOREM 3.3.1. *For the scaled workload vector $\overline{\boldsymbol{W}}_r$, we have*

$$\overline{\boldsymbol{W}}_r \stackrel{d}{\to} \hat{\boldsymbol{Z}}$$

as $r \to \infty$, where $\hat{\boldsymbol{Z}} = (\hat{Z}_1, \ldots, \hat{Z}_N)$, $\hat{Z}_i = \sup_{t\geq 0}\{Z_i(t)\}$ and $Z_i(t)$ is as defined in Section 3.2.

It is tempting to conclude directly from a combination of (3.3) and (3.7) that this theorem holds true by use of a continuous-mapping argument. However, complications arise since the supremum applied to càdlàg functions on the infinite domain $[0, \infty)$ is not necessarily a continuous functional. To prove Theorem 3.3.1, we have to justify the interchange of the heavy-traffic and the steady-state limits. To this end, observe that, as opposed to the infinite-domain case mentioned above, the supremum of càdlàg functions on a finite domain $[0, M)$, $M \in \mathbb{R}_+$, is a continuous functional (see e.g. [275]). The proof uses this fact in combination with an additional result stated in Lemma 3.3.4. To prove Lemma 3.3.4, we first establish upper bounds of the tail probabilities for the suprema of the processes $\{V_i(\lambda_{i,r} t) - \mathbb{E}[V_i(\lambda_{i,r})]t, t \geq 0\}$ and $\{\mathbb{E}[C_i(1)]t - C_i(t), t \geq 0\}$ in Lemmas 3.3.2 and 3.3.3, respectively.

LEMMA 3.3.2. *For the arrival process $\{V_i(\lambda_{i,r}), t \geq 0\}$ of Q_i, we have that*

$$\mathbb{P}\left(\sup_{t\in[0,T)} \{V_i(\lambda_{i,r} t) - \mathbb{E}[V_i(\lambda_{i,r})]t\} \geq x \right) \leq \frac{\lambda_{i,r} \mathbb{E}[B_i^2] T}{x^2}$$

for any $r, x, T \in \mathbb{R}_+$.

PROOF. As $\{V_i(\lambda_{i,r} t) - \mathbb{E}[V_i(\lambda_{i,r})]t, t \geq 0\}$ is a right-continuous martingale, we have by Doob's inequality (cf. [209, Theorem II.1.7]) that

$$\mathbb{P}\left(\sup_{t\in[0,T)} \{V_i(\lambda_{i,r} t) - \mathbb{E}[V_i(\lambda_{i,r})]t\} \geq x \right) \leq x^{-2} \sup_{t\in[0,T)} \{\mathrm{Var}[V_i(\lambda_{i,r} t)]\}.$$

Since $\mathrm{Var}[V_i(\lambda_{i,r} t)] = \lambda_{i,r} \sigma_{V,i}^2 t$ is strictly increasing in t, the lemma follows. □

LEMMA 3.3.3. *For the cumulative service process $\{C_i(t), t \geq 0\}$ pertaining to the server of Q_i, there exists, for every $x, T \in \mathbb{R}_+$, a set of positive real constants c_1, c_2, c_3 and c_4 such that*

$$\mathbb{P}\left(\sup_{t\in[0,T)} \{\mathbb{E}[C_i(1)]t - C_i(t)\} \geq x \right) \leq \frac{c_1 T}{x^2} + \frac{c_2}{T} + \frac{c_3 T}{e^{c_4\sqrt{x}}}.$$

PROOF. The lemma is a consequence of Proposition 1 in [131]. Define the constant $h = \max_{\omega\in\mathscr{S}}\{\phi_i(\omega)\}$ and the function $H(t) = ht - C_i(t)$. The process $\{H(t), t \geq 0\}$ represents increments of the regenerative process $\{h - \phi_i(\Phi(t)), t \geq 0\}$ and regenerates, for example, every time $\{\Phi(t), t \geq 0\}$ enters the reference state $\omega = \Phi(0)$. We denote the n-th of such regeneration times by T_n. Furthermore, we define $\gamma_n^* = \sup_{T_{n-1}\leq t\leq T_n}\{H(t) - H(T_{n-1})\}$ and $\nu_n = T_n - T_{n-1}$. Note that $\nu_1, \nu_2, \ldots$ can be seen as independent and identically distributed samples from a random variable Y and represent return times of state ω in the Markov chain $\{\Phi(t), t \geq 0\}$. Proposition 1 in [131] now implies that for all $x, T \in \mathbb{R}_+$, there exist positive real constants d_1, d_2, d_3 and d_4 such that

$$\mathbb{P}\left(\sup_{t\in[0,T)} \{\mathbb{E}[C_i(1)]t - C_i(t)\} > x \right) \leq d_1\left(e^{-d_2\frac{x^2}{T}} + e^{-d_3 T} + Te^{-d_4\sqrt{x}} \right) \tag{3.8}$$

if $\mathbb{E}[e^{\sqrt{\sup_{0\leq t\leq Y}\{H(t)\}}}] < \infty$ and $\mathbb{E}[e^{\sqrt{\gamma_n^*}}] < \infty$ for any $n \in \mathbb{N}_+$. This statement follows by replacing the variables B_t, b and $Q(x)$ in [131, Proposition 1] by $H(t)$, $h - \mathbb{E}[C_i(1)]$ and $\sqrt{x}$, respectively. To show that the necessary conditions hold in our case, observe that $H(t)$ is non-decreasing in t and takes values from $[0, ht]$. By combining this with the fact that $\sqrt{x} < \epsilon x + \frac{1}{\epsilon}$ for any $x \geq 0$ and $\epsilon > 0$, we have that $\mathbb{E}[e^{\sqrt{\sup_{0\leq t\leq Y}\{H(t)\}}}] = \mathbb{E}[e^{\sqrt{H(Y)}}] \leq \mathbb{E}[e^{\sqrt{hY}}] < \mathbb{E}[e^{\epsilon hY + \epsilon^{-1}}] = e^{\epsilon^{-1}}\mathbb{E}[e^{\epsilon hY}]$ for any $\epsilon > 0$. As $\gamma_n^* \leq h\nu_n$ for any $n > 0$, similar computations yield that $\mathbb{E}[e^{\sqrt{\gamma_n^*}}] < e^{\epsilon^{-1}}\mathbb{E}[e^{\epsilon hY}]$ for all $n \in \mathbb{N}$ and any $\epsilon > 0$. Subsequently, note that the regeneration time Y, which constitutes the return time of state ω in the Markov chain $\{\Phi(t), t \geq 0\}$, can be decomposed into a period of time Y_1 until the transition away from ω and the following period Y_2 until re-entry into state ω. The former period Y_1 is exponentially distributed with a certain rate α, so that $\mathbb{E}[e^{\epsilon hY_1}] = \frac{\alpha}{\alpha - \epsilon h}$ for $\epsilon < h^{-1}\alpha$. The latter period Y_2 is easily seen to be stochastically smaller than a geometrically distributed random variable with the positive success parameter $q = \min_{\omega'\in\mathscr{S}\setminus\{\omega\}}\{\mathbb{P}(\Phi(1) = \omega \mid \Phi(0) = \omega')\}$. Hence, $\mathbb{E}[e^{\epsilon hY_2}] \leq \frac{qe^{\epsilon h}}{1-(1-q)e^{\epsilon h}}$ for $\epsilon < -h^{-1}\log(1-q)$. As Y_1 and Y_2 are mutually independent, we thus have for $0 < \epsilon < h^{-1}\min\{\alpha, -\log(1-q)\}$ that $e^{\epsilon^{-1}}\mathbb{E}[e^{\epsilon hY}] \leq e^{\epsilon^{-1}}\frac{\alpha}{\alpha-\epsilon h}\frac{qe^{\epsilon h}}{1-(1-q)e^{\epsilon h}} < \infty$, so that the necessary conditions are satisfied. The lemma now follows from (3.8) by noting that $e^{-T} < T^{-1}$ for all $T > 0$ and taking $c_1 = d_1 d_2^{-1}, c_2 = d_1 d_3^{-1}, c_3 = d_1$ and $c_4 = d_4$. □

Based on the results obtained in Lemmas 3.3.2 and 3.3.3, we now establish the final auxiliary result needed to prove Theorem 3.3.1. This result is summarised in the following lemma.

LEMMA 3.3.4. *The scaled net-input process $\left\{\frac{V_i(\lambda_{i,r}r^2t) - C_i(r^2t)}{r}, t > 0\right\}$ corresponding to Q_i satisfies*

$$\lim_{M\to\infty}\lim_{r\to\infty}\mathbb{P}\left(\sup_{t\geq M}\left\{\frac{V_i(\lambda_{i,r}r^2t) - C_i(r^2t)}{r}\right\} \geq x \right) = 0$$

for all $x, M \in \mathbb{R}_+$.

PROOF. The first part of the proof is inspired by the proof of (20) in [223]. For any r, let $b_{i,r} = \frac{\mathbb{E}[V_i(\lambda_{i,r})]+\mathbb{E}[C_i(1)]}{2}$, so that $b_{i,r} - \mathbb{E}[V_i(\lambda_{i,r})] = \mathbb{E}[C_i(1)] - b_{i,r} = \frac{m_{C,i}-\lambda_{i,r}\mathbb{E}[B_i]}{2} = \beta_i m_{C,i}(2r)^{-1}$. Due to the subadditivity property of the supremum operator, we have for any $M > 0$ that

$$
\begin{aligned}
&\mathbb{P}\left(\sup_{t\geq M}\left\{\frac{V_i(\lambda_{i,r}r^2t)-C_i(r^2t)}{r}\right\}\geq x\right)\\
&\leq \mathbb{P}\left(\sup_{t\geq M}\left\{\frac{V_i(\lambda_{i,r}r^2t)-b_{i,r}r^2t}{r}\right\}+\sup_{t\geq M}\left\{\frac{b_{i,r}r^2t-C_i(r^2t)}{r}\right\}\geq x\right)\\
&\leq \mathbb{P}\left(\sup_{t\geq M}\{V_i(\lambda_{i,r}r^2t)-b_{i,r}r^2t\}\geq 0\right)+\mathbb{P}\left(\sup_{t\geq M}\{b_{i,r}r^2t-C_i(r^2t)\}\geq 0\right)\\
&\leq \sum_{j=0}^{\infty}\mathbb{P}\left(\sup_{t\in[2^jM,2^{j+1}M)}\{V_i(\lambda_{i,r}r^2t)-b_{i,r}r^2t\}\geq 0\right)\\
&\quad+\sum_{j=0}^{\infty}\mathbb{P}\left(\sup_{t\in[2^jM,2^{j+1}M)}\{b_{i,r}r^2t-C_i(r^2t)\}\geq 0\right)\\
&= \sum_{j=0}^{\infty}\mathbb{P}\left(\sup_{t\in[2^jr^2M,2^{j+1}r^2M)}\{V_i(\lambda_{i,r}t)-\mathbb{E}[V_i(\lambda_{i,r})]t-\beta_i m_{C,i}(2r)^{-1}t\}\geq 0\right)\\
&\quad+\sum_{j=0}^{\infty}\mathbb{P}\left(\sup_{t\in[2^jr^2M,2^{j+1}r^2M)}\{\mathbb{E}[C_i(1)]t-C_i(t)-\beta_i m_{C,i}(2r)^{-1}t\}\geq 0\right)\\
&\leq \sum_{j=0}^{\infty}\mathbb{P}\left(\sup_{t\in[0,2^{j+1}r^2M)}\{V_i(\lambda_{i,r}t)-\mathbb{E}[V_i(\lambda_{i,r})]t\}\geq 2^{j-1}\beta_i m_{C,i}rM\right)\\
&\quad+\sum_{j=0}^{\infty}\mathbb{P}\left(\sup_{t\in[0,2^{j+1}r^2M)}\{\mathbb{E}[C_i(1)]t-C_i(t)\}\geq 2^{j-1}\beta_i m_{C,i}rM\right)\\
&\leq \sum_{j=0}^{\infty}\frac{\lambda_{i,r}\mathbb{E}[B_i^2]2^{j+1}r^2M}{2^{2j-2}\beta_i^2m_{C,i}^2r^2M^2}\\
&\quad+\sum_{j=0}^{\infty}\left(\frac{c_1 2^{j+1}r^2M}{2^{2j-2}\beta_i^2m_{C,i}^2r^2M^2}+\frac{c_2}{2^{j+1}m_{C,i}r^2M}+\frac{c_3 2^{j+1}r^2M}{e^{c_4\sqrt{2^{j-1}\beta_i m_{C,i}rM}}}\right)
\end{aligned}
\tag{3.9}
$$

for certain positive constants c_1, c_2, c_3 and c_4. The penultimate inequality follows by observing that $\max_{t\in[2^jr^2M,2^{j+1}r^2M]}\{-\beta_i m_{C,i}(2r)^{-1}t\} = -2^{j-1}\beta_i m_{C,i}rM$ and by enlarging the intervals of the suprema to also include $[0,2^jr^2M)$. The last inequality follows from Lemmas 3.3.2 and 3.3.3. Simplifying (3.9) leads to

$$
\begin{aligned}
&\mathbb{P}\left(\sup_{t\geq M}\left\{\frac{V_i(\lambda_{i,r}r^2t)-C_i(r^2t)}{r}\right\}\geq x\right)\\
&\leq \frac{16(\lambda_{i,r}\mathbb{E}[B_i^2]+c_1)}{\beta_i^2m_{C,i}^2M}+\frac{c_2}{m_{C,i}r^2M}+\sum_{j=0}^{\infty}f_{i,j}(r,M),
\end{aligned}
\tag{3.10}
$$

where $f_{i,j}(r,M) = c_3 2^{j+1} r^2 M e^{-c_4\sqrt{2^{j-1}\beta_i m_{C,i} rM}}$. Observe that if

$$\lim_{r\to\infty} \sum_{j=0}^{\infty} f_{i,j}(r,M) = 0, \tag{3.11}$$

the lemma follows from (3.10) by taking the limit $r \to \infty$ and subsequently the limit $M \to \infty$ in (3.10). To show that the condition given in (3.11) indeed holds, observe that the derivative of $f_{i,j}$ with respect to r reads $\frac{\partial}{\partial r} f_{i,j}(r,M) = c_3 2^j r M e^{-h_{i,j}(M)\sqrt{r}}(4 - h_{i,j}(M)\sqrt{r})$, where $h_{i,j}(M) = c_4\sqrt{2^{j-1}\beta_i m_{C,i} M}$. As a result, $\frac{\partial}{\partial r} f_{i,j}(r,M) < 0$ if and only if $4 - h_{i,j}(M)\sqrt{r} < 0$. Due to the monotonicity of $h_{i,j}(M)$ and $\sqrt{r}$ in j and r, respectively, there thus exist positive constants j_0 and r_0, so that $\frac{\partial}{\partial r} f_{i,j}(r,M) < 0$ for any $j \geq j_0$ and $r \geq r_0$. This results in the fact that $\sup_{r\geq r_*} f_{i,j}(r,M) = f_{i,j}(r_*,M)$ for every $r_* \geq r_0$. Hence, an upper bound for $\sum_{j=0}^{\infty} f_{i,j}(r,M)$ when $r \geq r_* \geq r_0$ is given by

$$\sum_{j=0}^{\infty} f_{i,j}(r,M) = \sum_{j=0}^{j_0-1} f_{i,j}(r,M) + \sum_{j=j_0}^{\infty} f_{i,j}(r,M) \leq \sum_{j=0}^{j_0-1} f_{i,j}(r,M) + \sum_{j=j_0}^{\infty} f_{i,j}(r_*,M). \tag{3.12}$$

When $r \to \infty$, we can use (3.12) with r_* taken arbitrarily large so that

$$\lim_{r\to\infty} \sum_{j=0}^{\infty} f_{i,j}(r,M) \leq \lim_{r\to\infty} \sum_{j=0}^{j_0-1} f_{i,j}(r,M) + \sum_{j=j_0}^{\infty} \lim_{r_*\to\infty} f_{i,j}(r_*,M).$$

By observing that $\lim_{r\to\infty} f_{i,j}(r,M) = 0$, this reduces to $\lim_{r\to\infty} \sum_{j=0}^{\infty} f_{i,j}(r,M) \leq 0$. Hence, since $f_{i,j}(r,M) \geq 0$, it must hold that $\lim_{r\to\infty} \sum_{j=0}^{\infty} f_{i,j}(r,M) = 0$, which concludes the proof. □

Using these auxiliary results, we can now prove Theorem 3.3.1.

PROOF OF THEOREM 3.3.1. By (3.7), it is enough to show that

$$\lim_{r\to\infty} \mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\sup_{t\geq 0}\left\{\frac{V_i(\lambda_{i,r} r^2 t) - C_i(r^2 t)}{r}\right\} \geq x_i\right\}\right) = \mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\sup_{t\geq 0}\{Z_i(t)\} \geq x_i\right\}\right) \tag{3.13}$$

for all $x_1, \ldots, x_N \geq 0$. We first obtain a lower bound for the left-hand side of (3.13):

$$\begin{aligned}
&\lim_{r\to\infty} \mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\sup_{t\geq 0}\left\{\frac{V_i(\lambda_{i,r} r^2 t) - C_i(r^2 t)}{r}\right\} \geq x_i\right\}\right) \\
&\geq \lim_{r\to\infty} \mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\sup_{t\in[0,M)}\left\{\frac{V_i(\lambda_{i,r} r^2 t) - C_i(r^2 t)}{r}\right\} \geq x_i\right\}\right) \\
&= \mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\sup_{t\in[0,M)}\{Z_i(t)\} \geq x_i\right\}\right)
\end{aligned} \tag{3.14}$$

for all $M \in \mathbb{R}_+$, where the equality follows from (3.3) together with a combination of the continuous-mapping theorem and the continuity property of the supremum operator

applied to càdlàg-functions on the finite domain $[0,M)$. Next, to derive an upper bound for the left-hand side of (3.13), denote by $E_{M,i}$ the event that

$$\sup_{t\in[0,M)}\left\{\frac{V_i(\lambda_{i,r}r^2t)-C_i(r^2t)}{r}\right\}=\sup_{t\geq 0}\left\{\frac{V_i(\lambda_{i,r}r^2t)-C_i(r^2t)}{r}\right\},$$

and let $E_{M,i}^c$ be its complementary event. By De Morgan's law, we have that

$$\begin{aligned}
&\mathbb{P}\left(\bigcap_{i=1}^N\left\{\sup_{t\geq 0}\left\{\frac{V_i(\lambda_{i,r}r^2t)-C_i(r^2t)}{r}\right\}\geq x_i\right\}\right)\\
&=\mathbb{P}\left(\bigcap_{i=1}^N\left\{\sup_{t\geq 0}\left\{\frac{V_i(\lambda_{i,r}r^2t)-C_i(r^2t)}{r}\right\}\geq x_i;E_{M,i}\right\}\right)\\
&\quad+\mathbb{P}\left(\bigcap_{i=1}^N\left\{\sup_{t\geq 0}\left\{\frac{V_i(\lambda_{i,r}r^2t)-C_i(r^2t)}{r}\right\}\geq x_i\right\};\bigcup_{i=1}^N E_{M,i}^c\right).
\end{aligned}\tag{3.15}$$

An upper bound for the first term of the right-hand side in (3.15) is given by

$$\begin{aligned}
&\mathbb{P}\left(\bigcap_{i=1}^N\left\{\sup_{t\geq 0}\left\{\frac{V_i(\lambda_{i,r}r^2t)-C_i(r^2t)}{r}\right\}\geq x_i;E_{M,i}\right\}\right)\\
&\leq\mathbb{P}\left(\bigcap_{i=1}^N\left\{\sup_{t\in[0,M)}\left\{\frac{V_i(\lambda_{i,r}r^2t)-C_i(r^2t)}{r}\right\}\geq x_i\right\}\right)
\end{aligned}\tag{3.16}$$

for all $M\in\mathbb{R}_+$. For the second term of the right-hand side in (3.15), we have that

$$\begin{aligned}
&\mathbb{P}\left(\bigcap_{i=1}^N\left\{\sup_{t\geq 0}\left\{\frac{V_i(\lambda_{i,r}r^2t)-C_i(r^2t)}{r}\right\}\geq x_i\right\};\bigcup_{i=1}^N E_{M,i}^c\right)\\
&\leq\sum_{i=1}^N\mathbb{P}\left(\sup_{t\geq M}\left\{\frac{V_i(\lambda_{i,r}r^2t)-C_i(r^2t)}{r}\right\}\geq x_i\right),
\end{aligned}\tag{3.17}$$

for all $M\in\mathbb{R}_+$. Thus, by combining (3.15)–(3.17) and taking the limit $r\to\infty$, we obtain

$$\begin{aligned}
&\lim_{r\to\infty}\mathbb{P}\left(\bigcap_{i=1}^N\left\{\sup_{t\geq 0}\left\{\frac{V_i(\lambda_{i,r}r^2t)-C_i(r^2t)}{r}\right\}\geq x_i\right\}\right)\\
&\leq\mathbb{P}\left(\bigcap_{i=1}^N\left\{\sup_{t\in[0,M)}\{Z_i(t)\}\geq x_i\right\}\right)\\
&\quad+\lim_{r\to\infty}\sum_{i=1}^N\mathbb{P}\left(\sup_{t\geq M}\left\{\frac{V_i(\lambda_{i,r}r^2t)-C_i(r^2t)}{r}\right\}\geq x_i\right).
\end{aligned}\tag{3.18}$$

The lower bound established in (3.14) converges to $\mathbb{P}(\bigcap_{i=1}^N\{\sup_{t\in[0,\infty)}\{Z_i(t)\}\geq x_i\})$ as $M\to\infty$. The upper bound found in (3.18) also converges to this expression, as the second term in the right-hand side of (3.18) vanishes due to Lemma 3.3.4. From this, (3.13) immediately follows, which proves the theorem. □

REMARK 3.3.1. The joint distribution of $\hat{\boldsymbol{Z}}$ is not straightforward to derive explicitly. However, explicit expressions for the marginal distribution of $\hat{Z}_i$ are not hard to obtain. Note that $\hat{Z}_i = \sup_{t\geq 0} Z_i(t)$ is the all-time supremum of a one-dimensional Brownian motion with negative drift $-\beta_i m_{C,i}$ and variance $\frac{m_{C,i}}{\mathbb{E}[B_i]}\sigma_{V,i}^2+\sigma_{C,i}^2$. It is well known that the all-time supremum of a Brownian motion with negative drift $-a$ and variance b is exponentially $(\frac{2a}{b})$ distributed (cf. [19, Corollary IX.2.8 and Example IX.3.5]). Therefore, the distribution of the steady-state scaled workload $\overline{W}_{i,r}$ present in Q_i converges to an exponential distribution with rate $2\beta_i\left(\frac{\sigma_{V,i}^2}{\mathbb{E}[B_i]}+\frac{\sigma_{C,i}^2}{m_{C,i}}\right)^{-1}$ as $r\to\infty$. In the next section, we will see that the limiting distributions of $\overline{D}_{i,r}$ and $\overline{L}_{i,r}$ only differ from the limiting distribution of $\overline{W}_{i,r}$ by a multiplicative factor $m_{C,i}^{-1}$ and $\mathbb{E}[B_i]^{-1}$, respectively. As a result, the distributions of the steady-state delay $\overline{D}_{i,r}$ and the steady-state queue length $\overline{L}_{i,r}$ also converge to exponential distributions with rates $2\beta_i m_{C,i}\left(\frac{\sigma_{V,i}^2}{\mathbb{E}[B_i]}+\frac{\sigma_{C,i}^2}{m_{C,i}}\right)^{-1}$ and $2\beta_i\mathbb{E}[B_i]\left(\frac{\sigma_{V,i}^2}{\mathbb{E}[B_i]}+\frac{\sigma_{C,i}^2}{m_{C,i}}\right)^{-1}$, respectively. We study the derivation of the joint distribution of $\hat{\boldsymbol{Z}}$ in Section 3.5.2.

3.4 Extension to waiting times and queue lengths

In Section 3.3, we derived a heavy-traffic limit theorem for the scaled workload vector $\overline{\boldsymbol{W}}_r$. In this section, we extend this result to heavy-traffic limits for the distributions of the virtual waiting-time vector $\overline{\boldsymbol{D}}_r$ and the queue length vector $\overline{\boldsymbol{L}}_r$ by regarding the joint distribution of $\overline{\boldsymbol{D}}_r$ and $\overline{\boldsymbol{W}}_r$ as well as that of $\overline{\boldsymbol{L}}_r$ and $\overline{\boldsymbol{W}}_r$ in Section 3.4.1 and Section 3.4.2, respectively. It turns out that, when $r\to\infty$, the distributions of both $\overline{\boldsymbol{D}}_r$ and $\overline{\boldsymbol{L}}_r$ are elementwise equal to the distribution of $\overline{\boldsymbol{W}}_r$ up to a multiplicative constant.

3.4.1 Heavy-traffic asymptotics of the virtual waiting time

We now study the distribution of the scaled virtual waiting time in heavy traffic. First, we obtain the tail probability of the joint distribution of $\overline{\boldsymbol{D}}_r$ and $\overline{\boldsymbol{W}}_r$ as $r\to\infty$. Based on this, we obtain an extension of Theorem 3.3.1 for the scaled virtual waiting time.

PROPOSITION 3.4.1. *The tail probability of the limiting joint distribution of $\overline{\boldsymbol{D}}_r$ and $\overline{\boldsymbol{W}}_r$ satisfies*

$$\lim_{r\to\infty}\mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\overline{D}_{i,r}\geq s_i;\overline{W}_{i,r}\geq t_i\right\}\right)=\mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\hat{Z}_i\geq\max\{m_{C,i}s_i,t_i\}\right\}\right),$$

where $\hat{Z}_1,\ldots,\hat{Z}_N$ is defined as in Theorem 3.3.1.

PROOF. Observe that since the waiting time faced by an imaginary type-i customer arriving at time u is longer than s_i time units, the workload present in Q_i just before u is larger than $C_i(u+s_i)-C_i(u)$. This is evident, since the latter number represents the amount of work that the server of Q_i is able to process in the s_i time units following time u. In other words, the event $\{D_{i,r}(u)>s_i\}$ is tantamount to the event $\{W_{i,r}(u)>C_i(u+s_i)-C_i(u)\}$ for $i=1,\ldots,N$, so that in steady state (i.e. $u\to\infty$), we have

$$\mathbb{P}\left(\bigcap_{i=1}^{N}\left\{D_{i,r}>s_i;W_{i,r}>t_i\right\}\right)=\mathbb{P}\left(\bigcap_{i=1}^{N}\left\{W_{i,r}>\max\{C_i(s_i),t_i\}\right\}\right). \tag{3.19}$$

Based on this, we obtain an expression for the tail probability of the joint distribution of $\overline{\boldsymbol{D}}_r$ and $\overline{\boldsymbol{W}}_r$:

$$\mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\overline{D}_{i,r}\geq s_i;\overline{W}_{i,r}\geq t_i\right\}\right)=\mathbb{P}\left(\bigcap_{i=1}^{N}\left\{W_{i,r}\geq\max\{C_i(rs_i),rt_i\}\right\}\right)$$
$$=\mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\overline{W}_{i,r}\geq\max\left\{\frac{C_i(rs_i)}{r},t_i\right\}\right\}\right),\tag{3.20}$$

where we used (3.19) in the first equality. We now focus on showing that

$$\lim_{r\to\infty}\mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\overline{W}_{i,r}\geq\max\left\{\frac{C_i(rs_i)}{r},t_i\right\}\right\}\right)=\mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\hat{Z}_i\geq\max\{m_{C,i}s_i,t_i\}\right\}\right),\tag{3.21}$$

which combined with (3.20) directly implies the result to be proved. To this end, we observe that, since $\{C_i(t),t\geq 0\}$ is a renewal reward process, $r^{-1}C_i(rs_i)\to m_{C,i}s_i$ almost surely as $r\to\infty$ due to standard results in renewal theory. Denote by $F_{i,r}^{\epsilon}$ for any $\epsilon>0$ the event that $r^{-1}C_i(rs_i)\in[m_{C,i}s_i-\epsilon,m_{C,i}s_i+\epsilon]$. Thus, $\lim_{r\to\infty}\mathbb{P}(F_{i,r}^{\epsilon})=1$. As a result, we have due to De Morgan's law that

$$\mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\overline{W}_{i,r}\geq\max\left\{\frac{C_i(rs_i)}{r},t_1\right\}\right\}\right)=\mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\overline{W}_{i,r}\geq\max\left\{\frac{C_i(rs_i)}{r},t_i\right\};F_{i,r}^{\epsilon}\right\}\right)+o(1).$$

Letting $r\to\infty$ in this expression, using the definition of the event $F_{i,r}^{\epsilon}$ and applying Theorem 3.3.1, we obtain the following lower bound for the left-hand side of (3.21):

$$\lim_{r\to\infty}\mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\overline{W}_{i,r}\geq\max\left\{\frac{C_i(rs_i)}{r},t_i\right\}\right\}\right)\geq\mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\hat{Z}_i\geq\max\{m_{C,i}s_i+\epsilon,t_i\}\right\}\right).\tag{3.22}$$

Similarly, an upper bound for the left-hand side of (3.21) is given by

$$\lim_{r\to\infty}\mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\overline{W}_{i,r}\geq\max\left\{\frac{C_i(rs_i)}{r},t_i\right\}\right\}\right)\leq\mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\hat{Z}_i\geq\max\{m_{C,i}s_i-\epsilon,t_i\}\right\}\right).\tag{3.23}$$

In Remark 3.3.1, we found that $\hat{Z}_i$ is exponentially distributed for $i=1,\ldots,N$, so that the joint distribution of $\hat{\boldsymbol{Z}}$ has no discontinuity in the point $(m_{C,1}s_1,\ldots,m_{C,N}s_N)$. As a consequence, by taking the limit $\epsilon\to 0$ in the right-hand sides of (3.22) and (3.23), we obtain (3.21), which, as explained above, proves the proposition. □

From Proposition 3.4.1, the heavy-traffic limit for the virtual waiting time follows in the following corollary.

COROLLARY 3.4.2. *For the scaled virtual waiting-time vector $\overline{\boldsymbol{D}}_r$, it holds that*

$$\overline{\boldsymbol{D}}_r\xrightarrow{d}\left(\frac{1}{m_{C,1}},\ldots,\frac{1}{m_{C,N}}\right)\hat{\boldsymbol{Z}}$$

as $r\to\infty$, where $\hat{\boldsymbol{Z}}$ is defined as in Theorem 3.3.1.

PROOF. This follows immediately from Proposition 3.4.1 by taking $t_1=\cdots=t_N=0$. □

3.4.2 Heavy-traffic asymptotics of the joint queue length

In this section, we obtain an extension of Theorem 3.3.1 for the scaled steady-state queue length $\overline{\boldsymbol{L}}_r$ in heavy traffic. Let $B^R_{i,r}$ be the remaining service requirement of a type-i customer in service in the r-th system if $L_{i,r} > 0$, and zero otherwise. It is then trivially seen that

$$\boldsymbol{W}_r = (B^R_{1,r}, \ldots, B^R_{N,r}) + \left(\sum_{j=1}^{L_{1,r}} \widehat{B}_{1,j}, \ldots, \sum_{j=1}^{L_{N,r}} \widehat{B}_{N,j} \right) \tag{3.24}$$

for all $i > 0$, where $\widehat{B}_{i,j}$ represents the service requirement of the waiting customer in the j-th waiting position of Q_i and is distributed according to B_i. These service requirements are mutually independent as well as independent from $\boldsymbol{W}_r$ and $\boldsymbol{L}_r$. Note that $\widehat{B}_{i,j}$ is defined differently from $B_{i,j}$, which we defined in Section 3.2 to be the service requirement of the j-th arriving type-i customer since the start of the queueing process. The scaled version of (3.24) is given by

$$\overline{\boldsymbol{W}}_r = (\overline{B}^R_{1,r}, \ldots, \overline{B}^R_{N,r}) + \frac{1}{r} \left(\sum_{j=1}^{r\overline{L}_{1,r}} \widehat{B}_{1,j}, \ldots, \sum_{j=1}^{r\overline{L}_{N,r}} \widehat{B}_{N,j} \right), \tag{3.25}$$

where $\overline{B}^R_{i,r} = \frac{1}{r} B^R_{i,r}$ for $i = 1, \ldots, N$. It would intuitively be tempting to conclude that $(\overline{B}^R_{1,r}, \ldots, \overline{B}^R_{N,r}) \to \mathbf{0}$ as $r \to \infty$ and that as a result, $\overline{\boldsymbol{W}}_r$ and $\overline{\boldsymbol{L}}_r$ are equal elementwise up to a multiplicative constant. However, this is not straightforward, since, for example, $\overline{\boldsymbol{L}}_r$ and $(\overline{B}^R_{1,r}, \ldots, \overline{B}^R_{N,r})$ are not independent. We make these results rigorous in this section. Inspired by [295, Proposition 1], we first obtain another representation for the joint distribution of $\overline{L}_{i,r}$ and $\overline{W}_{i,r}$ for a single queue Q_i in Lemma 3.4.3. Based on this result, we derive the heavy-traffic asymptotics for $(\overline{L}_{i,r}, \overline{W}_{i,r}, \overline{B}^R_{i,r})$ in Lemma 3.4.4, which imply that $\overline{B}^R_{i,r} \to 0$ as $r \to \infty$. We subsequently conclude that $(\overline{B}^R_{1,r}, \ldots, \overline{B}^R_{N,r}) \to \mathbf{0}$ as $r \to \infty$ and derive the joint distribution of $\overline{\boldsymbol{L}}_r$ and $\overline{\boldsymbol{W}}_r$ as $r \to \infty$ in Proposition 3.4.5. From this, an extension of Theorem 3.3.1 for the scaled queue length $\overline{\boldsymbol{L}}_r$ follows in Corollary 3.4.6.

In order to construct an additional representation for the joint distribution of $\overline{L}_{i,r}$ and $\overline{W}_{i,r}$, we need to introduce some additional notation. Denote by $W^r_{i,n}$ and $L^r_{i,n}$ the workload present in Q_i and the queue length of Q_i, respectively, in the r-th system just before the n-th arrival of a type-i customer. Furthermore, $A^r_{i,j}$ refers to the time between the j-th and the $(j+1)$-st arriving type-i customer in the r-th system, so that $S^{A,r}_{i,n} = \sum_{j=1}^{n} A^r_{i,j}$ and $S^B_{i,n} = \sum_{j=1}^{n} B_{i,j}$ represent the cumulative series of interarrival times and service requirements of type-i customers. By construction of the heavy-traffic scaling, $A^r_{i,j} \xrightarrow{d} A_{i,j}$ and $\mathbb{E}[A^r_{i,j}] \to \mathbb{E}[A_{i,j}]$ as $r \to \infty$, where the random variables $A_{i,j}$ are independent and exponentially $\left(m_{C,i}/\mathbb{E}[B_i]\right)$ distributed. Finally, we define $S^r_{i,n} = S^B_{i,n} - C_i(S^{A,r}_{i,n})$. The required representation is now given in the following lemma.

LEMMA 3.4.3. *For any $x, y > 0$ and $i = 1, \ldots, N$, the joint distribution of $\overline{L}_{i,r}$ and $\overline{W}_{i,r}$*

satisfies

$$\mathbb{P}\big(\overline{L}_{i,r} \geq x; \overline{W}_{i,r} \geq y\big) = \mathbb{P}\bigg(W_{i,r} + B_i \geq C_i(S^{A,r}_{i,\lceil rx\rceil}); r^{-1}\max\Big\{W_{i,r} + S^r_{i,\lceil rx\rceil}, \max_{j\in\{1,\ldots,\lceil rx\rceil\}}\{S^r_{i,\lceil rx\rceil} - S^r_{i,j}\}\Big\} \geq y\bigg).$$

PROOF. The proof is inspired by [295, Proposition 1]. Observe that for any $k \geq 1$ and $n \geq 1$, the event $\{L^r_{i,n+k} \geq k\}$ coincides with the event that the workload that the server at Q_i was capable of processing between the arrival of the n-th and $(n+k)$-th customer, $C_i(S^{A,r}_{i,n+k-1}) - C_i(S^{A,r}_{i,n-1})$, does not exceed the amount $W^r_{i,n} + B_{i,n}$ of work present in Q_i just after the arrival of the n-th customer. Hence, we have that

$$\{L^r_{i,n+k} \geq k\} = \{W^r_{i,n} + B_{i,n} \geq C_i(S^{A,r}_{i,n+k-1}) - C_i(S^{A,r}_{i,n-1})\}. \tag{3.26}$$

Moreover, due to Lindley's recursion, which is given by $W^r_{i,n+1} = (W^r_{i,n} + S^r_{i,n} - S^r_{i,n-1})^+$ or

$$W^r_{i,n+k} = \max\Big\{W^r_{i,n} + S^r_{i,n+k-1} - S^r_{i,n-1}, \max_{j\in\{0,\ldots,k-1\}}\{S^r_{i,n+k-1} - S^r_{i,n+j}\}\Big\},$$

we have for any $y > 0$ that

$$\{W^r_{n+k} \geq y\} = \Big\{\max\Big\{W^r_{i,n} + S^r_{i,n+k-1} - S^r_{i,n-1}, \max_{j\in\{0,\ldots,k-1\}}\{S^r_{i,n+k-1} - S^r_{i,n+j}\}\Big\} \geq y\Big\}. \tag{3.27}$$

By combining (3.26) and (3.27), taking the probabilities of these events, letting $n \to \infty$ and observing that the vector $(L^r_{i,n}, W^r_{i,n})$ weakly converges to $(L_{i,r}, W_{i,r})$, we obtain

$$\mathbb{P}(L_{i,r} \geq k; W_{i,r} \geq y) = \mathbb{P}\bigg(W_{i,r} + B_i \geq C_i(S^{A,r}_{i,k}); \max\Big\{W_{i,r} + S^r_{i,k}, \max_{j\in\{1,\ldots,k\}}\{S^r_{i,k} - S^r_{i,j}\}\Big\} \geq y\bigg),$$

for any $k \geq 1, y > 0$. By noting that $\mathbb{P}(\overline{L}_{i,r} \geq x; \overline{W}_{i,r} \geq y) = \mathbb{P}(L_{i,r} \geq \lceil rx\rceil; r^{-1}W_{i,r} \geq y)$, the desired statement follows immediately. □

Based on Lemma 3.4.3, we derive the heavy-traffic asymptotics of $(\overline{L}_{i,r}, \overline{W}_{i,r}, \overline{B}^R_{i,r})$ in the following lemma. This lemma directly implies that $\overline{B}^R_{i,r} \to 0$ as $r \to \infty$.

LEMMA 3.4.4. *For any queue, the scaled steady-state queue length, workload and remaining service requirement exhibit state-space collapse under heavy-traffic assumptions. In particular, we have that*

$$(\overline{L}_{i,r}, \overline{W}_{i,r}, \overline{B}^R_{i,r}) \xrightarrow{d} \left(\frac{1}{\mathbb{E}[B_i]}, 1, 0\right)\hat{Z}_i$$

as $r \to \infty$ for any $i \in \{1,\ldots,N\}$, where $\hat{Z}_i$ is defined as in Section 3.2.

PROOF. Again, the proof is inspired by [295, Proposition 1]. We first focus on the joint distribution of $\overline{L}_{i,r}$ and $\overline{W}_{i,r}$. Due to the strong law of large numbers, $r^{-1}S^{A,r}_{i,\lceil rx\rceil} \to \mathbb{E}[A_{i,j}]x = \frac{\mathbb{E}[B_i]x}{m_{C,i}}$ almost surely as $r \to \infty$. Moreover, $t^{-1}C_i(t) \to m_{C,i}$ almost surely as $t \to \infty$, so that

$$\frac{C_i(S^{A,r}_{i,\lceil rx\rceil})}{r} = \frac{C_i(S^{A,r}_{i,\lceil rx\rceil})}{S^{A,r}_{i,\lceil rx\rceil}}\frac{S^{A,r}_{i,\lceil rx\rceil}}{r} \to \mathbb{E}[B_i]x \tag{3.28}$$

in probability as $r \to \infty$. We further have due to the weak law of large numbers that $r^{-1}S^B_{i,\lceil rx\rceil} \to \mathbb{E}[B_i]x$, so that $r^{-1}S^r_{i,\lceil rx\rceil} \to 0$ and $r^{-1}\max_{j\in\{1,\ldots,\lceil rx\rceil\}}\{S^r_{i,\lceil rx\rceil} - S^r_{i,j}\} \to 0$ as $r \to \infty$. For any $\epsilon > 0$, let $G^\epsilon_{i,r}$ denote the event

$$\{r^{-1}C_i(S^{A,r}_{i,\lceil rx\rceil}) \in [\mathbb{E}[B_i]x - \epsilon, \mathbb{E}[B_i]x + \epsilon]; r^{-1}S^B_{i,\lceil rx\rceil} \in [\mathbb{E}[B_i]x - \epsilon, \mathbb{E}[B_i]x + \epsilon];$$
$$r^{-1}S^r_{i,\lceil rx\rceil} \in [-\epsilon, \epsilon]; r^{-1}\max_{j\in\{1,\ldots,\lceil rx\rceil\}}\{S^r_{i,\lceil rx\rceil} - S^r_{i,j}\} \in [0, \epsilon]\}.$$

Due to the convergence results above, we have that $\lim_{r\to\infty}\mathbb{P}(G^\epsilon_{i,r}) = 1$, so that $\mathbb{P}(\overline{L}_{i,r} \geq x; \overline{W}_{i,r} \geq y) = \mathbb{P}(\overline{L}_{i,r} \geq x; \overline{W}_{i,r} \geq y; G^\epsilon_{i,r}) + o(1)$. After combining this with Lemma 3.4.3 and consequently taking the limit $r \to \infty$, we obtain

$$\lim_{r\to\infty}\mathbb{P}\left(\overline{W}_{i,r} \geq \max\{\mathbb{E}[B_i]x + \epsilon, y + \epsilon\}\right)$$
$$\leq \lim_{r\to\infty}\mathbb{P}\left(\overline{L}_{i,r} \geq x; \overline{W}_{i,r} \geq y\right) \leq \lim_{r\to\infty}\mathbb{P}\left(\overline{W}_{i,r} \geq \max\{\mathbb{E}[B_i]x - \epsilon, y - \epsilon\}\right),$$

since $\overline{B}_i \to 0$ as $r \to \infty$. By first applying Theorem 3.3.1 on the left-hand side and the right-hand side, next noting that the distribution of $\hat{Z}_i$ has no discontinuity points (cf. Remark 3.3.1) and finally letting $\epsilon \to 0$, we obtain

$$\lim_{r\to\infty}\mathbb{P}(\overline{L}_{i,r} \geq x; \overline{W}_{i,r} \geq y) = \mathbb{P}(\hat{Z}_i \geq \max\{\mathbb{E}[B_i]x, y\}). \tag{3.29}$$

It remains to consider the convergence of $\overline{B}^R_{i,r}$. We show that $\lim_{r\to\infty}\mathbb{P}(\overline{B}^R_{i,r} > \delta) = 0$ for all $\delta > 0$, which finalises the proof of the desired statement. Note that due to (3.25), we have that $\mathbb{P}(\overline{B}^R_{i,r} > \delta) = \mathbb{P}(\overline{W}_{i,r} > \frac{1}{r}\sum_{j=1}^{r\overline{L}_{i,r}}\widehat{B}_{i,j} + \delta)$. Let $H^\epsilon_{i,r}$ denote the event $\{\frac{1}{n}\sum_{j=1}^n \widehat{B}_{i,j} \in (\mathbb{E}[B_i] - \epsilon, \mathbb{E}[B_i] + \epsilon)\ \forall n \in [\sqrt{r}, \infty)\}$. By using the law of total probability and noting that $\lim_{r\to\infty}\mathbb{P}(H^\epsilon_{i,r}) = 1$ due to the weak law of large numbers, we thus have similar to earlier calculations that

$$\mathbb{P}(\overline{B}^R_{i,r} > \delta) = \mathbb{P}\left(\overline{W}_{i,r} > \frac{1}{r}\sum_{j=1}^{r\overline{L}_{i,r}}\widehat{B}_{i,j} + \delta; H^\epsilon_{i,r}\right) + o(1)$$
$$= \mathbb{P}\left(\overline{W}_{i,r} > \overline{L}_{i,r}\frac{1}{r\overline{L}_{i,r}}\sum_{j=1}^{r\overline{L}_{i,r}}\widehat{B}_{i,j} + \delta; H^\epsilon_{i,r}\right) + o(1).$$

By taking the limit $r \to \infty$ and using the established convergence of $\overline{L}_{i,r}$, we obtain

$$\lim_{r\to\infty}\mathbb{P}(\overline{W}_{i,r} > \overline{L}_{i,r}(\mathbb{E}[B_i] + \epsilon) + \delta) \leq \lim_{r\to\infty}\mathbb{P}(\overline{B}^R_{i,r} > \delta) \leq \lim_{r\to\infty}\mathbb{P}(\overline{W}_{i,r} > \overline{L}_{i,r}(\mathbb{E}[B_i] - \epsilon) + \delta).$$

By letting $\epsilon \to 0$ and noting, as before, that the limiting distribution of $\overline{W}_{i,r}$ has no discontinuity points, we have that $\lim_{r\to\infty}\mathbb{P}(\overline{B}^R_{i,r} > \delta) = \lim_{r\to\infty}\mathbb{P}(\overline{W}_{i,r} > \overline{L}_{i,r}\mathbb{E}[B_i] + \delta)$ for any $\delta > 0$. Observe that (3.29) implies that $\lim_{r\to\infty}\mathbb{P}(\overline{W}_{i,r} > \overline{L}_{i,r}\mathbb{E}[B_i] + \delta) = 0$ for any $\delta > 0$, which completes the proof. □

Based on the previous results, we now obtain the limiting joint distribution of $\overline{\boldsymbol{L}}_r$ and $\overline{\boldsymbol{W}}_r$ in the following proposition.

PROPOSITION 3.4.5. *The tail probability of the limiting joint distribution of* $\overline{\boldsymbol{L}}_r$ *and* $\overline{\boldsymbol{W}}_r$ *satisfies*

$$\lim_{r\to\infty} \mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\overline{L}_{i,r} \geq s_i; \overline{W}_{i,r} \geq t_i\right\}\right) = \mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\hat{Z}_i \geq \min\{\mathbb{E}[B_i]s_i, t_i\}\right\}\right), \tag{3.30}$$

where $\hat{Z}_1, \ldots, \hat{Z}_N$ *is defined as in Section 3.2.*

PROOF. Equation (3.25) implies that the event $\{\overline{L}_{i,r} \geq s_i\}$ coincides with the event $\{\overline{W}_{i,r} \geq \overline{B}^R_{i,r} + \frac{1}{r}\sum_{j=1}^{rs_i} \widehat{B}_{i,j}\}$, as the $\widehat{B}_{i,j}$ can only take non-negative values. Thus, we have

$$\mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\overline{L}_{i,r} \geq s_i; \overline{W}_{i,r} \geq t_i\right\}\right) = \mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\overline{W}_{i,r} \geq \max\left\{\overline{B}^R_{i,r} + \frac{1}{r}\sum_{j=1}^{rs_i}\widehat{B}_{i,j}, t_i\right\}\right\}\right).$$

Let $H^{\epsilon}_{i,r}$ be defined as before and recall that $\lim_{r\to\infty} \mathbb{P}(\bigcap_{i=1}^{N} H^{\epsilon}_{i,r}) = 1$, so that due to the law of total probability,

$$\begin{aligned}\mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\overline{L}_{i,r} \geq s_i; \overline{W}_{i,r} \geq t_i\right\}\right) = \mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\overline{W}_{i,r} \geq \max\left\{\overline{B}^R_{i,r} + s_i\frac{1}{rs_i}\sum_{j=1}^{rs_i}\widehat{B}_{i,j}, t_i\right\}; H^{\epsilon}_{i,r}\right\}\right) \\ + o(1).\end{aligned}$$

Note that according to Lemma 3.4.4, $\overline{B}^R_{i,r} \to 0$ as $r \to \infty$ for $i = 1, \ldots, N$, so that also $(\overline{B}^R_{1,r}, \ldots, \overline{B}^R_{N,r}) \to \mathbf{0}$ as $r \to \infty$. We thus obtain

$$\begin{aligned}\lim_{r\to\infty} \mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\overline{W}_{i,r} \geq \max\{\mathbb{E}[B_i] + \epsilon, t_i\}\right\}\right) &\leq \lim_{r\to\infty} \mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\overline{L}_{i,r} \geq s_i; \overline{W}_{i,r} \geq t_i\right\}\right) \\ &\leq \lim_{r\to\infty} \mathbb{P}\left(\bigcap_{i=1}^{N}\left\{\overline{W}_{i,r} \geq \max\{\mathbb{E}[B_i] - \epsilon, t_i\}\right\}\right).\end{aligned}$$

By taking the limit $\epsilon \to 0$, an application of Theorem 3.3.1 and the notion that the distribution of $\hat{\boldsymbol{Z}}$ has no discontinuity points yields the desired result. □

COROLLARY 3.4.6. *For the scaled queue length vector* $\overline{\boldsymbol{L}}_r$*, it holds that*

$$\overline{\boldsymbol{L}}_r \xrightarrow{d} \left(\frac{1}{\mathbb{E}[B_1]}, \ldots, \frac{1}{\mathbb{E}[B_N]}\right)\hat{\boldsymbol{Z}},$$

as $r \to \infty$*, where* $\hat{\boldsymbol{Z}}$ *is defined as in Section 3.2.*

PROOF. The desired statement follows immediately from Proposition 3.4.5 by taking $t_1 = \cdots = t_N = 0$. □

3.5 Application to the extended machine repair model

In this section, we apply the results obtained so far to the extended machine repair model. It is evident that this model with the model assumptions as stated in Section 2.2 fits the

framework of this chapter by taking $N = 2$, using $\{\mathbf{\Phi}(t), t \geq 0\} = \{(\Phi_1(t), \Phi_2(t)), t \geq 0\}$ as defined in Section 2.2 as the modulating Markov chain and choosing the state-dependent service speeds as $\phi_i(\omega) = \mathbb{1}_{\{\omega_i = U\}}$ for any ω in $\mathscr{S} = \{(U,U),(U,R),(R,U),(W,R),(R,W)\}$. Observe that the generator Q that corresponds to the modulating Markov chain $\{\mathbf{\Phi}(t), t \geq 0\}$ is now given by

$$Q = \begin{pmatrix} -\sigma_1 - \sigma_2 & \sigma_2 & \sigma_1 & 0 & 0 \\ \nu_2 & -\nu_2 - \sigma_1 & 0 & \sigma_1 & 0 \\ \nu_1 & 0 & -\nu_1 - \sigma_2 & 0 & \sigma_2 \\ 0 & 0 & \nu_2 & -\nu_2 & 0 \\ 0 & \nu_1 & 0 & 0 & -\nu_1 \end{pmatrix}.$$

We denote the elements of this matrix by $q_{i,j}$, $i,j \in \mathscr{S}$. Furthermore, we let $q_i = -q_{i,i}$ be the sum of the outgoing rates of state i. Recall that the invariant probability measure $\boldsymbol{\pi}$ is the unique solution of the equations $\boldsymbol{\pi} Q = \mathbf{0}$ and $\sum_{j \in \mathscr{S}} \pi_j = 1$.

In Section 3.5.1, we first study the remaining question of how to compute the covariance matrix $\mathbf{\Gamma}$ of the N-dimensional Brownian motion $\boldsymbol{Z}$. More specifically, we obtain expressions for the covariance terms $\gamma^C_{i,j}$ for the extended machine repair model by using results from the literature on Markov additive processes. We also compute the limiting distributions of $\overline{\boldsymbol{W}}_r$, $\overline{\boldsymbol{D}}_r$ and $\overline{\boldsymbol{L}}_r$. Doing so in an exact fashion turns out to be hard. Therefore, we study how to numerically obtain the limiting distributions by viewing $\hat{\boldsymbol{Z}}$ as an N-dimensional semi-martingale reflected Brownian motion in Section 3.5.2. Based on the resulting numerical computations, we conclude that the correlations between the first-layer queues of the extended machine repair model and thus also the interactions between the layers can be significant even in the heavy-traffic regime. Finally, in Section 3.5.3, we conclude by means of simulation that the distribution of $\overline{\boldsymbol{W}}_r$ converges quickly to the distribution of $\hat{\boldsymbol{Z}}$ as $r \to \infty$. Therefore, the heavy-traffic asymptotics constitute useful approximations for stable systems with a considerable load.

3.5.1 Derivation of the covariance matrix

We now demonstrate how to compute expressions for the covariance matrix $\mathbf{\Gamma}$ of the N-dimensional Brownian motion $\boldsymbol{Z}$ completely in terms of the model parameters. Although we do this based on the case of the extended machine repair model, the following methods can also be used to find the covariance matrix $\mathbf{\Gamma}$ for any instance of the generic model as described in Section 3.2 without any conceptual complications. By (3.4), it remains to compute expressions for the covariance terms $\gamma^C_{i,j} = \lim_{t \to \infty} \frac{1}{t} \mathrm{Cov}[C_i(t), C_j(t)]$ for all $i,j \in \{1,\ldots,N\}$. In order to compute these, observe that the increments of the processes $\{C_i(t), t \geq 0\}$ and $\{C_j(t), t \geq 0\}$ are conditionally independent given $\{\mathbf{\Phi}(t), t \geq 0\}$. Therefore, we can view $\{(\mathbf{\Phi}(t), C_i(t)), t \geq 0\}$, $\{(\mathbf{\Phi}(t), C_j(t)), t \geq 0\}$ and $\{(\mathbf{\Phi}(t), C_i(t) + C_j(t)), t \geq 0\}$ as Markov additive processes. For the definition and an overview of the standard theory on Markov additive processes, see [19, Section XI.2]. As a consequence, a functional central limit theorem for Markov additive processes obtained in [229] can be applied to compute $\gamma^C_{i,j}$ for all $i,j \in \{1,\ldots,N\}$. Let $\omega_{\mathrm{ref}} \in \mathscr{S}$ be an arbitrary reference state and let T_k be the k-th time after $t = 0$ that the Markov chain $\{\mathbf{\Phi}(t), t \geq 0\}$ enters this state. Then, the results of [229] imply the following lemma.

LEMMA 3.5.1. *Suppose that $\{Y(t), t \geq 0\}$ is a Markov-modulated drift process of which the drift equals d_k when the continuous-time Markov chain $\{\mathbf{\Phi}(t), t \geq 0\}$ is in state $k \in \mathscr{S}$.*

Furthermore, suppose that $|d_k| < \infty$ *for each* $k \in \mathscr{S}$ *and that* $\sum_{k\in\mathscr{S}} \pi_k d_k = 0$. *Then,* $\{\frac{1}{\sqrt{s}} Y(st), t \geq 0\}$ *converges in distribution, as* $s \to \infty$, *to a driftless Brownian motion starting at* 0 *with variance parameter*

$$\sigma_Y^2 = 2 \sum_{k\in\mathscr{S}} \pi_k \left(\frac{d_k^2}{q_k} + \sum_{l\in\mathscr{S}\setminus\{\{k\}\cup\{\omega_{ref}\}\}} \frac{q_{k,l} d_k f_l}{q_k} \right), \tag{3.31}$$

where the f_l*-parameters are the unique solution of the set of linear equations*

$$f_m = \frac{d_m}{q_m} + \sum_{n\in\mathscr{S}\setminus\{\{m\}\cup\{\omega_{ref}\}\}} \frac{q_{m,n}}{q_m} f_n.$$

In particular, we have that $\lim_{t\to\infty} \frac{1}{t}\mathrm{Var}[Y(t)] = \sigma_Y^2$.

PROOF. The convergence in distribution immediately follows from [229, Theorem 3.4] by taking $X(t) = \mathbf{\Phi}(t)$ and $D_{i,j} = V_{i,j} = v_i = 0$ for all $i, j \in \{1, \ldots, N\}$ in the notation of that paper. To show the result for the asymptotic variance of the modulated process Y, observe that $M(t) = \max_{k:T_k\leq t}\{k\}$ counts the number of times the Markov chain returned to the reference state up till time t, so that $\{M(t), t \geq 0\}$ can be interpreted as a (delayed) renewal process. As a consequence,

$$\begin{aligned}
\lim_{t\to\infty} \frac{\mathrm{Var}[Y(t)]}{t} &= \lim_{t\to\infty} \frac{\mathrm{Var}[Y(\sum_{i=1}^{M(t)}(T_{i+1} - T_i))] + o(t)}{t} \\
&= \lim_{t\to\infty} \frac{\mathbb{E}[M(t)]\mathrm{Var}[Y(T_2 - T_1)] + \mathrm{Var}[M(t)]\mathbb{E}[Y(T_2 - T_1)]^2}{t} \\
&= \mathrm{Var}[Y(T_2 - T_1)] \lim_{t\to\infty} \frac{\mathbb{E}[M(t)]}{t} = \frac{\mathrm{Var}[Y(T_2 - T_1)]}{\mathbb{E}[T_2 - T_1]}.
\end{aligned}$$

Section 3 in [229] shows that $\mathrm{Var}[Y(T_2 - T_1)] = \mathbb{E}[(Y(T_2 - T_1))^2] = \sigma_Y^2 \mathbb{E}[T_2 - T_1]$, which concludes the proof. □

We now apply this lemma to obtain the covariance matrix that corresponds to the extended machine repair model. In particular, to compute $\sigma_{C,1}^2$, we study the process

$$Y(t) = C_1(t) - \mathbb{E}[C_1(t)] = C_1(t) - (\pi_{(U,U)} + \pi_{(U,R)})t$$

with conditional drift $d_k = \mathbb{1}_{\{k\in\{(U,U),(U,R)\}\}} - (\pi_{(U,U)} + \pi_{(U,R)})$ when the modulating process $\{\mathbf{\Phi}(t), t \geq 0\}$ resides in state k. As $\mathrm{Var}[Y(t)] = \mathrm{Var}[C_1(t)]$ for any $t \geq 0$, an expression for $\sigma_{C,1}^2$ is then readily given in Lemma 3.5.1 by (3.31). An expression for $\sigma_{C,2}^2$ can be found similarly to the computations above or simply by interchanging the indices in the found expression for $\sigma_{C,1}^2$. Observe that an expression for $\lim_{t\to\infty} \frac{1}{t}\mathrm{Var}[C_1(t) + C_2(t)]$ can also be found using the same technique, but now considering the process

$$\begin{aligned}
Y(t) &= C_1(t) + C_2(t) - (\mathbb{E}[C_1(t) + C_2(t)]) \\
&= C_1(t) + C_2(t) - (2\pi_{(U,U)} + \pi_{(U,R)} + \pi_{(R,U)})t
\end{aligned}$$

instead with $d_k = \mathbb{1}_{\{k\in\{(U,U),(U,R)\}\}} + \mathbb{1}_{\{k\in\{(U,U),(R,U)\}\}} - (2\pi_{(U,U)} + \pi_{(U,R)} + \pi_{(R,U)})$. Again, it then holds that an expression for $\lim_{t\to\infty} \frac{1}{t}\mathrm{Var}[C_1(t) + C_2(t)]$ is given in (3.31). After

these computations, the covariance matrix $\mathbf{\Gamma}$ can be expressed explicitly in terms of the model parameters. The covariance parameters $\gamma^C_{1,1}$ and $\gamma^C_{2,2}$ are by definition equal to $\sigma^2_{C,1}$ and $\sigma^2_{C,2}$, for which we have already derived explicit expressions. As for the remaining parameters, we have that both $\gamma^C_{1,2}$ and $\gamma^C_{2,1}$ are equal to

$$\begin{aligned}&\lim_{t\to\infty}\frac{1}{t}\mathrm{Cov}[C_1(t),C_2(t)]\\&=\frac{1}{2}\left(\lim_{t\to\infty}\frac{1}{t}\mathrm{Var}[C_1(t)+C_2(t)]-\lim_{t\to\infty}\frac{1}{t}\mathrm{Var}[C_1(t)]-\lim_{t\to\infty}\frac{1}{t}\mathrm{Var}[C_2(t)]\right),\end{aligned}$$

where all of the terms between the brackets in the right-hand side are now known. As the rest of the terms appearing in (3.4) were already expressed in terms of the model parameters, the covariance matrix $\mathbf{\Gamma}$ is now explicitly known.

3.5.2 Numerical evaluation of the limiting distribution of $\hat{\boldsymbol{Z}}$

Now that $\mathbf{\Gamma}$ can be computed explicitly, we investigate in this section the joint distribution of $\hat{\boldsymbol{Z}}$, the limiting distribution of the scaled workload $\overline{\boldsymbol{W}}_r$, in stationarity. Since the limiting distributions of $\overline{\boldsymbol{D}}_r$ or $\overline{\boldsymbol{L}}_r$ equal the distribution of $\hat{\boldsymbol{Z}}$ up to a scalar as observed in Corollaries 3.4.2 and 3.4.6, the results also directly relate to the limiting distributions of the scaled virtual waiting time and the scaled queue length.

To study the joint distribution of $\hat{\boldsymbol{Z}}$ as defined in Theorem 3.3.1, we first observe that this distribution equals the stationary distribution of an N-dimensional semi-martingale reflected Brownian motion. In particular, by the definitions of $\boldsymbol{Z}(t)$ and $\hat{Z}_i(t)$ in Section 3.2 and Theorem 3.3.1, respectively, we have that the process $\hat{\boldsymbol{Z}}(t)=\{\hat{Z}_1(t),\dots,\hat{Z}_N(t)\}$ satisfies

$$\begin{aligned}\hat{\boldsymbol{Z}}(t)&=\left(\sup_{s\in[0,t]}\{Z_1(s)\},\dots,\sup_{s\in[0,t]}\{Z_N(s)\}\right)\\&\overset{d}{=}\left(\sup_{s\in[0,t]}\{Z_1(t)-Z_1(t-s)\},\dots,\sup_{s\in[0,t]}\{Z_N(t)-Z_N(t-s)\}\right)\\&=\left(Z_1(t)-\inf_{s\in[0,t]}\{Z_1(s)\},\dots,Z_N(t)-\inf_{s\in[0,t]}\{Z_N(s)\}\right)\\&=\boldsymbol{Z}(t)+R\boldsymbol{Y}(t),\end{aligned}$$

where the equality in distribution follows since multi-dimensional Brownian motions are time-reversible [32, Lemma II.2]. In this representation, R is the $N\times N$ identity matrix and $\boldsymbol{Y}(t)=(Y_1(t),\dots,Y_N(t))=(-\inf_{s\in[0,t]}\{Z_1(s)\},\dots,-\inf_{s\in[0,t]}\{Z_N(s)\})$. Observe that $\{\boldsymbol{Y}(t),t\geq 0\}$ is a continuous, non-decreasing process starting in $\mathbf{0}$, of which the elements Y_i can only increase at times t when $\hat{Z}_i(t)=0$. A process with such a representation is known to be a semi-martingale reflected Brownian motion on the state space $\mathbb{R}^N_+$ (see e.g. [60, Section 7.4]). By letting $t\to\infty$, it is now clear that the joint distribution of $\hat{\boldsymbol{Z}}$ coincides with the stationary distribution of a semi-martingale reflected Brownian motion on the non-negative orthant with drift vector $\boldsymbol{\mu}$, covariance matrix $\mathbf{\Gamma}$ and reflection matrix R.

In general, the computation of the stationary distribution of a multi-dimensional semi-martingale reflected Brownian motion is a challenging problem. Although the semi-

TABLE 3.1: Numerical results for several instances of the extended machine repair model.

	Parameters										Results		
Instance no.	β_1	β_2	$\mathbb{E}[B_1]$	$\mathbb{E}[B_1^2]$	$\mathbb{E}[B_2]$	$\mathbb{E}[B_2^2]$	σ_1	σ_2	ν_1	ν_2	$\mathbb{E}[\hat{Z}_1]$	$\mathbb{E}[\hat{Z}_2]$	$\mathrm{Corr}[\hat{Z}_1, \hat{Z}_2]$
1	1	1	1	2	1	2	$\frac{1}{10}$	$\frac{1}{10}$	$\frac{1}{10}$	$\frac{1}{10}$	4.33	4.33	0.274
2	$\frac{1}{2}$	1	1	2	1	2	$\frac{1}{10}$	$\frac{1}{10}$	$\frac{1}{10}$	$\frac{1}{10}$	8.67	4.33	0.228
3	1	1	1	5	1	5	$\frac{1}{10}$	$\frac{1}{10}$	$\frac{1}{10}$	$\frac{1}{10}$	5.83	5.83	0.195
4	1	1	$\frac{1}{2}$	$\frac{1}{2}$	2	8	$\frac{1}{5}$	$\frac{1}{20}$	$\frac{1}{5}$	$\frac{1}{20}$	3.84	7.18	0.446
5	1	1	1	2	1	2	1	1	1	1	1.33	1.33	0.080
6	1	1	1	2	1	2	$\frac{1}{20}$	$\frac{1}{20}$	$\frac{1}{5}$	$\frac{1}{5}$	2.06	2.06	0.124

martingale reflected Brownian motion corresponding to our model satisfies the conditions derived in [119] for a unique stationary distribution to exist, it does not necessarily satisfy the necessary requirements found in [120] for this distribution to have a product form. Nonetheless, a numerical approach obtained in [70] to compute the stationary distribution is applicable to our setting.

We now apply this numerical algorithm to the extended machine repair model and observe several parameter effects. Observe that for the extended machine repair model, R resolves to a 2×2 identity matrix and that the underlying Brownian motion $\{\boldsymbol{Z}(t), t \geq 0\}$ has a drift vector

$$\boldsymbol{\mu} = \big(-\beta_1(\pi_{(U,U)} + \pi_{(U,R)}), -\beta_2(\pi_{(U,U)} + \pi_{(R,U)})\big)$$

and a covariance matrix

$$\boldsymbol{\Gamma} = diag\left(\frac{\mathbb{E}[B_1^2]}{\mathbb{E}[B_1]}(\pi_{(U,U)} + \pi_{(U,R)}), \frac{\mathbb{E}[B_2^2]}{\mathbb{E}[B_2]}(\pi_{(U,U)} + \pi_{(R,U)})\right) + \boldsymbol{\Gamma}^C,$$

where $\boldsymbol{\Gamma}^C$ is a 2×2 matrix consisting of the elements $\gamma^C_{i,j}$ computed in Section 3.5.1. For a number of instances of the extended machine repair model, we have computed several characteristics of the stationary distribution, such as the first two moments and the cross-moment of $\hat{Z}_1$ and $\hat{Z}_2$. The results are summarised in Table 3.1, where for each of the instances the found values for $\mathbb{E}[\hat{Z}_1]$, $\mathbb{E}[\hat{Z}_2]$ and the correlation coefficient

$$\mathrm{Corr}[\hat{Z}_1, \hat{Z}_2] = \frac{\mathbb{E}[\hat{Z}_1\hat{Z}_2] - \mathbb{E}[\hat{Z}_1]\mathbb{E}[\hat{Z}_2]}{\sqrt{\mathbb{E}[\hat{Z}_1^2] - \mathbb{E}[\hat{Z}_1]^2}\sqrt{\mathbb{E}[\hat{Z}_2^2] - \mathbb{E}[\hat{Z}_2]^2}}$$

are given. Recall that the marginal distribution of $\hat{Z}_i$ is exponential, so that $\mathbb{E}[\hat{Z}_i^2] = 2\mathbb{E}[\hat{Z}_i]^2$. Observe also that the limiting distributions of $\overline{\boldsymbol{D}}_r$ and $\overline{\boldsymbol{L}}_r$ are equal to the distribution of $\hat{\boldsymbol{Z}}$ up to a scalar, so that $\mathrm{Corr}[\hat{Z}_1, \hat{Z}_2]$ does not only represent the correlation coefficient pertaining to the limiting distribution of the scaled workload $\overline{\boldsymbol{W}}_r$, but also to that of the scaled virtual waiting time and the scaled queue length. It follows from Table 3.1 that the competition between the machines of the repair facilities can be of

TABLE 3.2: Simulation results for $\overline{\boldsymbol{W}}_5, \overline{\boldsymbol{W}}_{10}$ and $\overline{\boldsymbol{W}}_{20}$.

	Results								
Instance no.	$\mathbb{E}[\overline{W}_{1,5}]$	$\mathbb{E}[\overline{W}_{1,10}]$	$\mathbb{E}[\overline{W}_{1,20}]$	$\mathbb{E}[\overline{W}_{2,5}]$	$\mathbb{E}[\overline{W}_{2,10}]$	$\mathbb{E}[\overline{W}_{2,20}]$	$\text{Corr}[\overline{W}_{1,5}, \overline{W}_{2,5}]$	$\text{Corr}[\overline{W}_{1,10}, \overline{W}_{2,10}]$	$\text{Corr}[\overline{W}_{1,20}, \overline{W}_{2,20}]$
1	3.46	3.90	4.12	3.46	3.90	4.12	0.262	0.271	0.273
2	7.80	8.23	8.45	3.46	3.90	4.12	0.217	0.225	0.228
3	4.42	5.11	5.47	4.42	5.11	5.47	0.180	0.189	0.192
4	3.08	3.46	3.65	5.72	6.46	6.82	0.466	0.460	0.453
5	1.07	1.20	1.27	1.07	1.20	1.27	-0.053	0.001	0.044
6	1.64	1.85	1.95	1.64	1.85	1.95	0.121	0.126	0.125

such a level that the correlation coefficient pertaining to the queue lengths is significant. Moreover, by taking the first instance as a reference, we observe that the correlation coefficient is highly influenced by the relative convergence speed of the arrival rates (instance no. 2), the variability of the service times (instance no. 3), the level of asymmetry in the model parameters (instance no. 4), the frequency of machine breakdowns and speed of machine repairs with respect to the arrivals and services of products (instance no. 5), and the duration of the machine's uptimes with respect to that of their repairs (instance no. 6).

3.5.3 Comparison with simulation results

We end this section with an assessment of the quality of the distribution of $\hat{\boldsymbol{Z}}$ as an approximation for the joint workload distribution in systems with a considerable load. In Table 3.2, simulation results for the scaled workload $\overline{\boldsymbol{W}}_r$ corresponding to the values $r = 5, 10, 20$ are given for each of the instances given in Table 3.1. Recall that $\rho_i = 1 - \frac{\beta_i}{r}$, so that $r = 5, 10, 20$ corresponds to $\rho_i = 0.8, 0.9, 0.95$ if $\beta_i = 1$. Thus, the values $r = 5, 10, 20$ represent systems that operate under a high load, as is often the case in practice.

As expected, Tables 3.1 and 3.2 suggest that the distribution of $\hat{\boldsymbol{Z}}$ generally approximates the distribution of $\overline{\boldsymbol{W}}_r$ well in terms of marginal means and the correlation coefficient. In particular, the tables confirm that $\mathbb{E}[\overline{W}_{i,r}]$ converges to $\mathbb{E}[\hat{Z}_i]$ from below as $r \to \infty$ at a fast rate, so that $\mathbb{E}[\hat{Z}_i]$ is a provably useful upper bound close to the actual value of $\mathbb{E}[\overline{W}_{i,r}]$ for large r (i.e. significantly loaded systems). Surprisingly, the rate at which $\mathbb{E}[\overline{W}_{i,r}]$ converges to $\mathbb{E}[\hat{Z}_i]$ does not seem to differ much between the model instances. The slowest convergence occurs in the third model instance due to the high variability of the service times, but it does not deviate much from the other instances. The only outlying rate of convergence can be found in the expected scaled waiting time of the first queue in the second model instance, where convergence is a lot faster. However, this is obvious by the nature of our scaling, since $\beta_1 = 1/2$ for that model instance instead of

$\beta_1 = 1$.

Furthermore, the values of $\text{Corr}[\hat{Z}_1, \hat{Z}_2]$ given in Table 3.1 turn out to be accurate approximations of the values $\text{Corr}[W_{1,r}, W_{2,r}]$ given in Table 3.2 for almost all of the model instances and any $r \in \{5, 10, 20\}$. Thus, the limiting distribution seems to capture the correlation structure between the queue lengths in the stable case rather well. One can argue that the fifth model instance is an exception to this. However, due to the high frequency of machine breakdowns and repairs, there hardly is any correlation between the queues, making correlation coefficients hard to approximate accurately.

4

CLOSED-FORM APPROXIMATIONS FOR EXPECTED QUEUE LENGTHS

In this chapter, we construct two closed-form approximations for the expected queue length of any first-layer queue in the extended machine repair model by using the light-traffic and the heavy-traffic results derived in Chapters 2 and 3, respectively. The first approximation is based only on the light-traffic asymptotics, and we show through a numerical study that this approximation already performs surprisingly well for arbitrarily loaded systems. Refinement of this approximation using the heavy-traffic behaviour of the queue length distribution leads to a second approximation, which remains in closed form, and its accuracy seems to be on par with that of numerical methods. These approximations may prove to be very useful for optimisation purposes due to their accuracy and their closed-form property.

4.1 Introduction

Based on the findings of Chapters 2 and 3, we now propose two approximations for the mean queue lengths of the first-layer queues in the extended machine repair model. In this chapter, we will present approximations for the queue length of Q_1, the first queue of products, but similar results for Q_2, the second queue of products, are readily obtained by interchanging indices. The first approximation is based on the light-traffic behaviour of the mean queue length as studied in Chapter 2. More specifically, we assume that $\mathbb{E}[L_1]$, the mean queue length of Q_1, can be seen as an analytic function of $\hat{\rho}_1$ in $[0,1)$, and we choose this function such that its derivatives near $\hat{\rho}_1 = 0$ are in line with the coefficients $f_1(0), f_1(1)$ and $f_1(2)$ as computed in Section 2.4.1 by the power-series algorithm, i.e. the first few coefficients of $f(k)$ in (2.10) corresponding to $g(\boldsymbol{l}, \boldsymbol{\varphi}) = l_1$ and $\chi = \hat{\rho}_1$ up to $k = 2$. As we will see, this approximation already achieves a very good accuracy. Moreover, since the coefficients $f_1(0), f_1(1)$ and $f_1(2)$ are known explicitly, the approximation can be expressed in closed form. Therefore, it is easily implementable and suitable for optimisation purposes.

In an effort to further increase accuracy, we derive a second approximation, which is also consistent with the heavy-traffic theorems obtained in Chapter 3. In principle,

the idea behind this refined approximation is to interpolate between the derived light-traffic and heavy-traffic asymptotics based on the value of $\hat{\rho}_1$. In the literature (see e.g. [99, 206, 273]), such interpolation approximations have been proposed in the past to approximate performance measures in the GI/G/1 queue and in queueing systems with Poisson input. More recently, a similar interpolation approximation has been applied successfully to approximate the mean waiting times in polling systems with renewal arrivals [45], which has acted as a basis for a distributional waiting-time approximation in such systems [P9]. Interpolation approximations derived in the spirit of these papers are also often well-suited for optimisation purposes due to their simple form, as is demonstrated in [P10].

The interpolation approximation that we derive in this chapter is still in closed form and works even better than the first approximation in terms of accuracy, being indistinguishable from numerical results.

In the remainder of this chapter, we will use the model assumptions and the notation introduced in Section 2.2. In Section 4.2, we derive the first approximation based on the light-traffic asymptotics of the mean queue length and show by a numerical study that it performs very well over a wide range of parameter settings. Subsequently, in Section 4.3, we derive the second approximation, which also incorporates the correct heavy-traffic behaviour. Finally, Section 4.4 presents a number of limiting cases of the model where the approximation turns out to be exact.

4.2 Light-traffic approximation

In this section, we derive a light-traffic approximation for $\mathbb{E}[L_1]$, the mean queue length of Q_1. The approximation, which we denote by $\mathbb{E}[L_{1,app}^{LT}]$, is based on the symbolic closed-form expressions $f_1(0)$, $f_1(1)$ and $f_1(2)$. We also numerically assess its accuracy.

4.2.1 Derivation

To derive an approximation for $\mathbb{E}[L_1]$, recall that $\hat{\rho}_1 = \frac{\lambda_1}{\mu_1 m_{C,1}}$ represents the level of saturation of Q_1. We consider the mean queue length of Q_1 as a function h of $\hat{\rho}_1$. We assume this function to be analytic on $[0,1)$. In other words, we assume that this function can be written as

$$h(\hat{\rho}_1) = \sum_{n=0}^{\infty} f_1(n)\hat{\rho}_1^n = \frac{z(\hat{\rho}_1)}{1-\hat{\rho}_1} \tag{4.1}$$

for $0 \leq \hat{\rho}_1 < 1$, where

$$f_1(n) = \frac{h^{(n)}(0)}{n!},\ z(\hat{\rho}_1) = f_1(0) + \sum_{n=1}^{\infty}(f_1(n) - f_1(n-1))\hat{\rho}_1^n \tag{4.2}$$

and $h^{(n)}(0)$ is the n-th derivative of h with respect to $\hat{\rho}_1$ evaluated at $\hat{\rho}_1 = 0$. Observe that the power series (2.10) and (4.1) are equal when taking $g(\boldsymbol{l}, \boldsymbol{\varphi}) = l_1$ and $\chi = \hat{\rho}_1$. As a consequence, although an exact expression for $h(\hat{\rho}_1)$ is not known, the coefficients $f_1(n)$, $n = 0, 1, \ldots$ can be computed using the computational scheme as given in Section 2.3.2. In Section 2.4.1, we have already obtained symbolic closed-form expressions

for $f_1(0) = 0, f_1(1)$ and $f_1(2)$ in the model parameters $\mu_1, \sigma_1, \sigma_2, \nu_1$ and ν_2. Since $h(\hat{\rho}_1)$ is guaranteed to exist, we approximate the value of $z(\hat{\rho}_1)$. When numerically observing the first few terms of the series $\{f_1(i) - f_1(i-1), i > 0\}$ using the computational scheme of Section 2.3.2, we generally see that they are moderate in absolute value, but more importantly, alternate in sign. This even seems to be the case when this series is divergent. Because of this and the decreasing nature of $\hat{\rho}_1^n$ in n, we may assume that the first two terms alone already approximate this sum well. In other words, since $f_1(0)$ equals zero, we have that $z(\hat{\rho}_1)$ should be well approximated by $f_1(1)\hat{\rho}_1 + (f_1(2) - f_1(1))\hat{\rho}_1^2$. From this observation, a light-traffic approximation for $\mathbb{E}[L_1]$ follows immediately.

APPROXIMATION 4.2.1. *In the extended machine repair model, a closed-form approximation for the mean queue length of Q_1 is given by*

$$\mathbb{E}[L_{1,app}^{LT}] = \frac{a\hat{\rho}_1 + b\hat{\rho}_1^2}{1 - \hat{\rho}_1}, \tag{4.3}$$

where $a = f_1(1)$ and $b = f_1(2) - f_1(1)$. The coefficients $f_1(1)$ and $f_1(2)$ are computed in Section 2.4.1.

An extensive numerical study in the next section shows that Approximation 4.2.1 performs very well in terms of accuracy. Furthermore, because the approximation is given in a simple and closed form, it is very easy to implement and suitable for optimisation purposes.

4.2.2 Accuracy

To numerically assess the accuracy of Approximation 4.2.1, we apply the light-traffic approximation to a number of systems and compare it to values of the mean queue length of Q_1 obtained by numerical methods. The complete test bed of instances that we analysed contains 675 different combinations of parameter values, all listed in Table 4.1. This table lists multiple values for the normalised load of Q_1 (i.e. $\hat{\rho}_1$), the breakdown rates of M_1 and M_2 (i.e. σ_1 and σ_2) and the repair rates of M_1 and M_2 (i.e. ν_1 and ν_2). In particular, these rates are varied in the order of magnitude through the values a_i^σ and a_i^ν, and in the imbalance through the values b_j^σ and b_j^ν, as specified in the table. As a consequence, the breakdown rates (σ_1, σ_2) and the repair rates (ν_1, ν_2) run from $(0.1, 0.1)$, being small and perfectly balanced, to $(50, 10)$, being large and significantly imbalanced. The service requirements of type-1 products are assumed to be exponentially (1) distributed.

For each of the model instances corresponding to each of the parameter combinations in Table 4.1, we compare $\mathbb{E}[L_{1,app}^{LT}]$, the *approximated* mean queue length of Q_1, to numerically computed values of $\mathbb{E}[L_1]$, the mean queue length of Q_1. In most cases, we have computed the numerical values for $\mathbb{E}[L_1]$ using the power-series algorithm numerically with $M = 39$. In these cases, the power series in (2.10) converges and thus produces values with high precision in less time than simulation would (although the time needed is still significant). The numerical error made in truncating this power series can then be estimated by computing $\sum_{i=M+1}^{\infty} \chi^k f(M)$, which evaluated to a number less than 2×10^{-5} times the actual computed value for $\mathbb{E}[L_1]$ in the worst case, and is on average much smaller. For some cases where the uptimes and repair times are on average much longer than the interarrival and service times, lengthy simulation runs were used to compute the numerical values.

TABLE 4.1: Parameter values of the test bed used to compare the light-traffic approximation to numerical results.

Parameter	Considered parameter values
ρ_1	$\{0.25, 0.5, 0.75\}$
μ_1	$\{1\}$
(σ_1, σ_2)	$a_i^\sigma \cdot b_j^\sigma \quad \forall i, j,$ where $\boldsymbol{a}^\sigma = \{0.1, 1, 10\}$ and $\boldsymbol{b}^\sigma = \{(1,1),(1,2),(2,1),(1,5),(5,1)\}$
(ν_1, ν_2)	$a_i^\nu \cdot b_j^\nu \quad \forall i, j,$ where $\boldsymbol{a}^\nu = \{0.1, 1, 10\}$ and $\boldsymbol{b}^\nu = \{(1,1),(1,2),(2,1),(1,5),(5,1)\}$

Subsequently, we compute the relative error of these approximations. In other words, for every instance of the testbed, we compute

$$\Delta = 100\% \times \left| \frac{\mathbb{E}[L_{1,app}^{LT}] - \mathbb{E}[L_1]}{\mathbb{E}[L_1]} \right|.$$

The average value of the errors corresponding to the instances is roughly 0.05%. The largest error encountered in this test bed has an average value of $\Delta = 1.72\%$ and corresponds to the system with model parameters $\hat{\rho}_1 = 0.75$ and $\sigma_1 = \sigma_2 = \nu_1 = \nu_2 = 0.1$. This is a system for which the breakdowns and repairs occur on the slowest time scale compared to the interarrival times and service times of the products in the first queue.

In Table 4.2, the mean values of Δ are given for each category of the variables in Table 4.1. We see in Table 4.2(a) that the accuracy of the approximation increases as the load offered to Q_1 decreases. This is not surprising, as the approximation is exact in light traffic by construction. From Tables 4.2(b) and 4.2(c), it is clear that the approximation is sensitive to the magnitude of the breakdown rates and repair rates. As will become evident in Section 4.4, the approximation becomes exact as some of these variables tend to zero or infinity. Moreover, according to Tables 4.2(d) and 4.2(e), the approximation is less sensitive to imbalance in the second layer of the system.

Based on these results, we conclude that the approximation works very well in general. The accuracy may degrade slightly when breakdown rates and repair rates are very small compared to the arrival and service rate of type-1 products. To illustrate this, regard a system with $\mu_1 = 1$ and $\sigma_1 = \sigma_2 = \nu_1 = \nu_2 = 0.001$. In Figure 4.1, we plot the light-traffic approximation $\mathbb{E}[L_{1,app}^{LT}]$ for this system along with numerical values for $\mathbb{E}[L_1]$, both as a function of $\hat{\rho}_1$. In this extreme example, Δ grows up to roughly 6% as $\hat{\rho}_1$ nears one. However, the light-traffic approximation remains very well suited for optimisation purposes. The shapes of the curves of $\mathbb{E}[L_{1,app}^{LT}]$ and $\mathbb{E}[L_1]$ still match each other well. Therefore, using the derived light-traffic approximation in an optimisation function instead of an exact expression if it had been available, should result in an optimum that is close to the true optimum.

TABLE 4.2: Mean percentual relative error Δ categorised in ρ_1 (a), a_i^σ (b), a_i^ν (c), b_j^σ (d) and b_j^ν (e).

(a)

$\hat{\rho}_1$	0.25	0.5	0.75
Mean rel. error Δ	0.01%	0.05%	0.10%

(b)

a_i^σ	0.1	1	10
Mean rel. error Δ	0.15%	0.01%	0.00%

(c)

a_i^ν	0.1	1	10
Mean rel. error Δ	0.15%	0.01%	0.00%

(d)

b_j^σ	(1, 1)	(1, 2)	(2, 1)	(1, 5)	(5, 1)
Mean rel. error Δ	0.07%	0.06%	0.07%	0.03%	0.04%

(e)

b_j^ν	(1, 1)	(1, 2)	(2, 1)	(1, 5)	(5, 1)
Mean rel. error Δ	0.07%	0.06%	0.03%	0.04%	0.07%

4.3 Interpolation approximation

Approximation 4.2.1 satisfies the light-traffic limits found by the power-series algorithm, and we have seen that it already performs very well for arbitrarily loaded systems. Nevertheless, the accuracy degrades slightly as $\hat{\rho}_1$ nears one. To increase the performance in this region, we refine the approximation so that it also satisfies known heavy-traffic behaviour. More specifically, we will now also require that the approximation, as $\hat{\rho}_1$ approaches one, coincides with the mean of the limiting queue length distribution as computed in Section 3. The refined approximation, which we denote by $\mathbb{E}[L_{1,app}^{IP}]$, interpolates between the light-traffic and heavy-traffic limits on the basis of $\hat{\rho}_1$, and we will hence also refer to it as the interpolation approximation. To derive this approximation, we again assume the form $\mathbb{E}[L_{1,app}^{IP}] = \frac{r(\hat{\rho}_1)}{1-\hat{\rho}_1}$, where $r(\hat{\rho}_1)$ is a polynomial function in $\hat{\rho}_1$. Note that this form is in line with previously derived interpolation approximations in the literature [45, 99, 206, 273].

Recall that in Approximation 4.2.1, $r(\hat{\rho}_1)$ was chosen to be a second-order polynomial. Now that we have the additional requirement of satisfying heavy-traffic behaviour, we choose $r(\hat{\rho}_1)$ to be a third-order polynomial. In short, we impose the following constraints

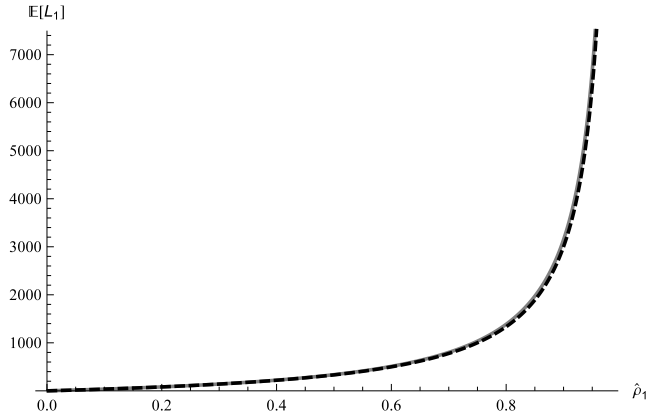

FIGURE 4.1: $\mathbb{E}[L^{LT}_{1,app}]$ (solid curve) and $\mathbb{E}[L_1]$ (dashed curve) as a function of $\hat{\rho}_1$.

on the interpolation approximation. First, we require the approximated mean waiting time at $\hat{\rho}_1 = 0$ and its first two derivatives with respect to $\hat{\rho}_1$ evaluated at the same point to be equal to the corresponding exact values obtained by the power-series algorithm:

1. $\mathbb{E}[L^{IP}_{1,app}]|_{\hat{\rho}_1=0} = \mathbb{E}[L_1]|_{\hat{\rho}_1=0} = f_1(0) = 0$,
2. $\frac{d}{d\hat{\rho}_1}\mathbb{E}[L^{IP}_{1,app}]|_{\hat{\rho}_1=0} = \frac{d}{d\hat{\rho}_1}\mathbb{E}[L_1]|_{\hat{\rho}_1=0} = f_1(1)$,
3. $\frac{d^2}{d\hat{\rho}_1^2}\mathbb{E}[L^{IP}_{1,app}]|_{\hat{\rho}_1=0} = \frac{d^2}{d\hat{\rho}_1^2}\mathbb{E}[L_1]|_{\hat{\rho}_1=0} = 2f_1(1) + 2f_1(2)$.

Moreover, we require the interpolation approximation to coincide with the mean of the heavy-traffic limiting distribution of the queue length. By taking $\beta_1 = 1$ in the framework of Chapter 3 and recalling that service times are exponentially (μ_1) distributed, we have by Remark 3.3.1 that in heavy traffic, the (scaled) queue length of Q_1 is exponentially distributed with mean $1 + \frac{\mu_1\sigma^2_{C,1}}{2m_{C,1}}$. Thus, we require that

4. $\lim_{\hat{\rho}_1\uparrow 1}\mathbb{E}[(1-\hat{\rho}_1)L^{IP}_{1,app}] = 1 + \frac{\mu_1\sigma^2_{C,1}}{2m_{C,1}}$.

Recall that we already computed a closed-form expression for the variance parameter $\sigma^2_{C,1}$ in Section 3.5.1, so that this mean is completely known. The assumptions and constraints above now fully determine the following approximation.

APPROXIMATION 4.3.1. *In the extended machine repair model, a closed-form approximation for the mean queue length of Q_1 is given by*

$$\mathbb{E}[L^{IP}_{1,app}] = \frac{a\hat{\rho}_1 + b\hat{\rho}_1^2 + c\hat{\rho}_1^3}{1-\hat{\rho}_1}, \tag{4.4}$$

where $a = f_1(1)$, $b = f_1(2) - f_1(1)$ and $c = 1 + \frac{\mu_1\sigma^2_{C,1}}{2m_{C,1}} - f_1(2)$. The coefficients $f_1(1), f_1(2)$ and the variance parameter $\sigma^2_{C,1}$ are computed in Section 2.4.1 and Section 3.5.1, respectively.

We end this section by observing that Approximation 4.3.1 performs extremely well in terms of accuracy. When comparing results of this interpolation approximation for each of the cases displayed in Table 4.1 to the numerical values computed in Section 4.2.2 to

inspect the accuracy of the light-traffic approximation, we find that the size differences are of the same order as the expected accuracy error of the numerical methods. However, the computational effort needed to apply the interpolation approximation is, due to its closed-form nature, much less than that of any numerical method.

As the accuracy of the interpolation approximation seems to be comparable to that of numerical methods, it is hard to observe any possible parameter effects. Nevertheless, several conjectures can be made about the sensitivity of the accuracy of Approximation 4.3.1 to the model parameters. For example, as the approximation satisfies light-traffic and heavy-traffic results, its accuracy is assumed to be best for $\hat{\rho}_1$ values close to zero or one. Furthermore, as the interpolation approximation includes the ingredients used to construct the light-traffic approximation, it is reasonable to assume that the accuracy of the interpolation approximation is also sensitive to the magnitude of the breakdown and repair rates similarly to Approximation 4.2.1.

REMARK 4.3.1. We proposed Approximations 4.2.1 and 4.3.1 for a model with two queues and one repairman. However, similar strategies to those used in this section lead to accurate approximations for models with larger numbers of queues and repairmen. To obtain the light-traffic terms a and b, the implementation of the power-series algorithm must be adapted, as suggested in Remark 2.3.4. For the heavy-traffic term, the results from Chapter 3 still apply. However, expressions for $m_{C,1}$ and $\sigma^2_{C,1}$ must be recomputed based on the adapted cumulative service process $\{C_1(t), t \geq 0\}$. Similarly, when relaxing the model to allow for phase-type distributed service times, breakdown times and repair times, we can still apply the power-series algorithm to obtain light-traffic results, as explained in Remark 2.3.3. As for the heavy-traffic term, again only the expressions for $m_{C,1}$ and $\sigma^2_{C,1}$ have to be recomputed.

4.4 Behaviour in asymptotic regimes

We conclude this chapter by commenting on the behaviour of Approximation 4.2.1 and Approximation 4.3.1 in asymptotic instances of the extended machine repair model.

Light traffic and heavy traffic By construction, both the light-traffic and the interpolation approximations are exact for systems where Q_1 is lightly loaded, i.e. systems where λ_1 tends to zero. Furthermore, the interpolation approximation coincides with the mean of the limiting distribution of the scaled queue length $(1-\hat{\rho}_1)L_1$ when $\hat{\rho}_1$ tends to one. The latter property is highly desirable from a practical perspective, as one is often interested in cases where the queues are heavily loaded. For example, in manufacturing, one is typically interested in maximising the utilisation of the machines without significantly deteriorating the performance of the system.

No M_1-breakdowns In case M_1 never breaks down (i.e. $\sigma_1 = 0$), both the light-traffic approximation and the interpolation approximation are exact. When there are no M_1-breakdowns, Q_1 behaves like a regular M/M/1 queue. For the M/M/1 model, it is known that

$$\mathbb{E}[L_1] = \sum_{n=0}^{\infty} \hat{\rho}_1^{n+1} = \frac{\hat{\rho}_1}{1-\hat{\rho}_1}. \tag{4.5}$$

Since M_1 never breaks down, we obviously have that $m_{C,1} = 1$ and $\sigma^2_{C,1} = 0$. Moreover, we have that $f_1(1)|_{\sigma_1=0} = f_1(2)|_{\sigma_1=0} = 1$. Therefore, it is easy to see that (4.3), (4.4) and (4.5) coincide when there are no M_1-breakdowns.

No M_2-breakdowns or instant M_2-repairs In case M_2 does not require any repair time from the repairman, both approximations are exact as well. Downtimes of M_1 then only consist of the actual repair times and are exponentially (ν_1) distributed. Let the completion time C of a type-1 product be the time between the start of its service period and the moment it leaves the system. It is easily verified that Q_1 in isolation can be modelled as an M/G/1 queue with server vacations starting at epochs when the queue becomes empty. We refer to this vacation queue as Y. We obtain the expected queue length of Q_1 in this limiting regime by studying the mean queue length $\mathbb{E}[L_Y]$ of the equivalent vacation queue Y. The service times in Y correspond to the completion times in Q_1, and the vacation times in Y are composed of the idle times of M_1 plus the downtimes corresponding to breakdowns that occurred when there was no product in Q_1. Due to the Fuhrmann-Cooper decomposition property [102] applied to Y, the mean queue length of Y can be decomposed as follows:

$$\mathbb{E}[L_Y] = \mathbb{E}[L_{M/G/1}] + \mathbb{E}[L_Y | Y \text{ in vacation period}]. \tag{4.6}$$

The first term in the right-hand side corresponds to the expected queue length in an M/G/1 queue similar to Y, but where the server does not incur any vacations. The second term is the mean queue length in Y observed at a point in time at which the server is on vacation. By standard methods, we find after some trivial computations that

$$\mathbb{E}[L_Y | Y \text{ in vacation period}] = \frac{\lambda_1}{\nu_1} \frac{\sigma_1}{\sigma_1 + \nu_1}.$$

This result is not surprising, as this expression equals the mean number of Poisson arrivals during a past part of a downtime D_1, which is exponentially (ν_1) distributed, times the probability $\frac{\sigma_1}{\sigma_1+\nu_1}$ that a product arriving in an empty system finds the machine not in an operational state, but in need of repair.

Furthermore, it is well known that

$$\mathbb{E}[L_{M/G/1}] = \lambda_1 \mathbb{E}[C] + \frac{\lambda_1^2 \mathbb{E}[C^2]}{2(1 - \lambda_1 \mathbb{E}[C])}.$$

The moments $\mathbb{E}[C]$ and $\mathbb{E}[C^2]$ of the completion time can be determined by using the relation $C = B_1 + \sum_{i=1}^{N(B_1)} V_i$, where V_i is the duration of the i-th downtime incurred within the completion time C. The random variable $N(B_1)$ denotes the number of breakdowns during the service period B_1 and is Poisson $(\sigma_1 B_1)$ distributed. The downtimes V_i are exponentially (ν_1) distributed, as a downtime now only consists of a single repair time. This relation leads to the following Laplace-Stieltjes transform of the completion time:

$$\begin{aligned}\mathbb{E}[e^{-sC}] = \mathbb{E}[e^{-s(B_1 + \sum_{i=1}^{N(B_1)} V_i)}] &= \int_{t=0}^{\infty} e^{-st} \mathbb{E}[e^{-s\sum_{i=1}^{N(t)} V_i}] d\mathbb{P}(B_1 < t) \\ &= \int_{t=0}^{\infty} e^{-st} \left(\sum_{x=0}^{\infty} \mathbb{E}[e^{-sV_1}]^x e^{-\sigma_1 t} \frac{(\sigma_1 t)^x}{x!} \right) d\mathbb{P}(B_1 < t)\end{aligned}$$

$$= \mathbb{E}[e^{-\left(s+\sigma_1(1-\mathbb{E}[e^{-sV_1}])\right)B_1}] = \frac{\mu_1}{\mu_1 + s + \sigma_1(1 - \frac{\nu_1}{\nu_1+s})},$$

out of which the moments of C follow by differentiating with respect to s and substituting $s = 0$:

$$\mathbb{E}[C] = \frac{\nu_1 + \sigma_1}{\mu_1 \nu_1} \text{ and } \mathbb{E}[C^2] = \frac{2\left(\mu_1\sigma_1 + (\nu_1 + \sigma_1)^2\right)}{\mu_1^2 \nu_1^2}.$$

Since M_2 requires no repair time, we have that $m_{C,1} = \frac{\nu_1}{\sigma_1+\nu_1}$ and $\hat{\rho}_1 = \frac{\lambda_1}{\mu_1}\frac{\sigma_1+\nu_1}{\nu_1}$. By combining the results above,

$$\mathbb{E}[L_1] = \frac{\left(1 + \frac{\sigma_1\mu_1}{(\sigma_1+\nu_1)^2}\right)\hat{\rho}_1}{1-\hat{\rho}_1}. \tag{4.7}$$

One can show that $f_1(1)|_{\sigma_2=0} = f_1(2)|_{\sigma_2=0} = f_1(1)|_{\nu_2\to\infty} = f_1(2)|_{\nu_2\to\infty} = 1 + \frac{\sigma_1\mu_1}{(\sigma_1+\nu_1)^2}$. Since (4.7) is also exact in the limit $\hat{\rho}_1 \to 1$, the heavy-traffic term $1 + \frac{\mu_1\sigma_{C,1}^2}{2m_{C,1}}$ also equals this value. Because of these observations, (4.3), (4.4) and (4.7) coincide whenever there are no M_2-breakdowns or M_2-repairs are instant.

5

APPROXIMATIONS FOR THE COMPLETE QUEUE LENGTH DISTRIBUTION

This chapter aims to find approximations for the *complete* (marginal) queue length distributions of the first-layer queues in the extended machine repair model. We do so by drawing a connection between a first-layer queue and a single-server queue with correlated server downtimes. Based on a careful study of the second layer of the extended machine repair model, we make an explicit assumption on the form of the dependence between the consecutive downtimes of a machine, which holds approximately. We analyse the complete queue length distribution of the single-server queue with this downtime structure and use the results to approximate the queue length distributions in the extended machine repair model. By means of a numerical study, we subsequently show this approximation to be highly accurate.

5.1 Introduction

To approximate the complete queue length distribution of a first-layer queue, we regard this queue as a single-server queue in isolation. In Section 1.3.1, we observed that the consecutive downtimes of each machine exhibit autocorrelation. Therefore, we model the first-layer queue as an M/G/1 queue with interdependent vacation lengths in order to capture these correlations. More specifically, we use the following approach:

1. For the single-server queue, we use an explicit, generic dependence form for the vacation lengths and obtain approximate yet very accurate results for the queue length distribution.
2. For the extended machine repair model, we compute several characteristics of the downtime structure, such as the first two moments of the downtime distribution and the correlation coefficient of the consecutive downtimes of a machine.
3. We choose the parameters of the generic dependence form of the single-server model so that they match the downtime characteristics of the extended machine repair model computed in the previous step.

Thus, we use the results from step one with the parameters from step two as an approximation for the marginal queue length distributions of the first-layer queue.

As mentioned in Section 1.3.1, a similar approach has been used by Wartenhorst in [269] to derive approximations for the first two moments of the queue length distribution. In that study, Wartenhorst assumes exponential service times, and equal uptime and repair-time distributions for the machines. These are assumptions that we generalise in this chapter. Wartenhorst subsequently approximates the first two moments of a first-layer queue by computing those of a single-server vacation queue where the distribution of the vacation lengths is taken to be equal to that of the machine's downtimes in the extended machine repair model, but the vacation lengths are assumed to be *completely independent*. The resulting approximation is exact by construction for a system where downtimes are independent and accurate whenever downtimes are only slightly dependent. Since the dependence is completely ignored, this approximation becomes more inaccurate as the dependence increases. In this chapter, we explicitly model the dependence, thus improving accuracy greatly, and obtain an approximation for the *complete distribution* of the queue length.

The M/G/1 queue with server vacations has been studied extensively; see e.g. [84, 85] for surveys. Often, vacation lengths or downtimes are assumed to be independent of any other event in the system. Exceptions can be found in [118], where vacation lengths are dependent on the number of customers in the system, and in [53], where vacation lengths are dependent on the length of the previous active period of the server. In the context of polling systems, vacation queues with interdependent vacation lengths have been considered in [18, 93, 108]. However, in that context, the start of a server vacation is usually confined to a point in time at which the server concludes the service of a customer. This is not the case in the current context, where a machine can break down at any point in time.

The rest of this chapter is structured as follows. Section 5.2 provides in detail the model assumptions that we use in this chapter to study the extended machine repair model and introduces the single-server model, its dependence structure and all of the notation required. In Section 5.3, we analyse the queue length distribution of the single-server queue at various time epochs. This results in an approximate expression for the (probability generating function of the) steady-state queue length distribution at an arbitrary point in time. We believe this result to be of independent interest, but our main goal is to apply this result to the extended machine repair model. By connecting both models, the approximate expression for the queue length distribution of the single-server queue also leads to an approximation for the marginal queue length distribution of the corresponding first-layer queue in the extended machine repair model. The latter approximation forms the main result of this chapter and is discussed in Section 5.4. Finally, Section 5.5 provides extensive numerical results showing that the obtained approximation is highly accurate and identifies the factors determining the level of accuracy.

5.2 Model description and notation

In this section, we state our model assumptions for the extended machine repair model, and we introduce the single-server queue with its specific dependence structure, along with all the necessary notation.

The extended machine repair model When it concerns the extended machine repair model, we mostly follow the model assumptions and notation as introduced in Sections 1.3.1 and 2.2. We only deviate from the previous assumptions when it concerns the service times of the products. In this chapter, we assume the service times of type-i products to be generally distributed according to some random variable B_i. The Laplace-Stieltjes transform $\mathbb{E}[e^{-sB_i}]$ corresponding to this random variable is denoted by $\widetilde{B}_i(s)$. The load offered to Q_i is then defined as $\rho_i = \lambda_i \mathbb{E}[B_i]$. Note that Q_i is stable if $\rho_i < \frac{\mathbb{E}[U_i]}{\mathbb{E}[U_i]+\mathbb{E}[D_i]}$. Finally, we assume a *pre-emptive repeat* policy: when a machine breaks down, the service of a product in progress is aborted and will be restarted once the machine is operational again.

The single-server model In the single-server model, the queue is fed by a Poisson process with parameter λ. The service time B required by arriving customers is generally distributed. The uptime U from the moment a server has just ended a vacation period until the start of the next one is exponentially distributed with parameter σ. After this time period U, the server starts a vacation for D time units (a downtime). If a job is in service when the server starts a vacation, all of the work done on the job is lost and processing of the job is restarted once the server ends its vacation (*pre-emptive repeat*). The Laplace-Stieltjes transform corresponding to the service time, $\mathbb{E}[e^{-sB}]$, is denoted by $\widetilde{B}(s)$. Likewise, the downtime D is represented by the Laplace-Stieltjes transform $\widetilde{D}(s) = \mathbb{E}[e^{-sD}]$. The steady-state queue length of the queue, including the job in service, is denoted by L. It will prove convenient to regard the queue length distribution at specific time epochs. To this end, let M and N denote the queue length at the beginning and the end of an arbitrary downtime, respectively.

This model differs from most vacation queues studied in literature, because in our case the durations of vacations (or breakdowns) are dependent. In particular, we assume these durations to be one-dependent; i.e. we assume that the duration of a vacation directly depends on the duration of the preceding vacation. Given the duration of the preceding vacation, however, it does not depend on even earlier vacations. We use a generic dependence structure that can be used to model positive correlations between consecutive downtimes. We describe the dependence structure of the downtimes by specifying the Laplace-Stieltjes transform of a downtime $D(k+1)$ conditioned on its previous downtime $D(k)$:

$$\mathbb{E}[e^{-sD(k+1)}|D(k) = t] = \chi(s)e^{-g(s)t}, \tag{5.1}$$

where $\chi(s)$ and $g(s)$ are analytic functions in s with $\chi(0) = 1 - g(0) = 1$. This generic dependence structure is introduced in [47] to model positive correlation between two random variables. It can be interpreted as follows. The downtime $D(k+1)$ consists of an independent component, which is represented by the Laplace-Stieltjes transform $\chi(s)$, and a component dependent on the previous downtime, which is represented by $e^{-g(s)t}$. In particular, if one assumes that $g(s)$ has a completely monotone derivative (i.e. $(-1)^{n+1}\frac{d^n}{ds^n}g(s) \geq 0$ for all $n \geq 1$), then $e^{-g(s)}$ is the Laplace-Stieltjes transform of an infinitely divisible distribution (see [96, p. 450]). We will use this assumption in the proof of Lemma 5.3.1.

To give an indication of how rich the class of dependence structures that satisfy (5.1) is, note that the class of infinitely divisible distributions is strongly connected to the class of Lévy processes (see e.g. [154, Chapter 1]). In particular, for a Lévy process $\{X(t), t \geq 0\}$,

one has $\mathbb{E}[e^{-sX(t)}] = e^{-g(s)t}$, where $g(s)$ is a function with a completely monotone derivative. Thus, $D(k+1)$ consists of a time component independent from the previous downtime $D(k)$ and another component, the value of which is that of a Lévy process observed at a time which is governed by $D(k)$. For several examples of dependence structures that (5.1) covers, see e.g. [47] or [267].

In the extended machine repair model, a downtime can also be thought of as the sum of an independent component (e.g. the repair time) and a component dependent on the previous downtime (the waiting time). Therefore, the functions $\chi(s)$ and $g(s)$ can be chosen in such a way that they together represent the distribution and the dependence of these downtimes closely. As we discuss in Section 5.4, (5.1) does not model the downtimes of the extended machine repair model perfectly. However, as we will see in Section 5.5, it is a good fit.

Note that the functions $\chi(s)$ and $g(s)$ determine the stationary downtime D. In particular, in stationarity it holds that $\mathbb{E}[e^{-sD(k+1)}] = \mathbb{E}[e^{-sD(k)}] = \widetilde{D}(s)$, so we have that

$$\widetilde{D}(s) = \int_{t=0}^{\infty} \chi(s) e^{-g(s)t} d\mathbb{P}(D < t) = \chi(s)\widetilde{D}(g(s)). \tag{5.2}$$

As a result, the first two moments are given by

$$\begin{aligned} \mathbb{E}[D] &= -\widetilde{D}'(0) = \frac{\chi'(0)}{g'(0)-1} \quad \text{and} \\ \mathbb{E}[D^2] &= \widetilde{D}''(0) = \frac{\chi''(0) - \mathbb{E}[D](2\chi'(0)g'(0) + g''(0))}{1 - g'(0)^2}. \end{aligned} \tag{5.3}$$

By iterating (5.2), one obtains an explicit expression for $\widetilde{D}(s)$:

$$\widetilde{D}(s) = \prod_{j=0}^{\infty} \chi(g^{(j)}(s)), \tag{5.4}$$

where $g^{(0)}(s) = s$ and $g^{(j)}(s) = g(g^{(j-1)}(s))$. The bivariate Laplace-Stieltjes transform of $D(k)$ and $D(k+1)$ is given by

$$\begin{aligned} \mathbb{E}[e^{-s_1 D(k) - s_2 D(k+1)}] &= \int_{t=0}^{\infty} e^{-s_1 t} \mathbb{E}[e^{-s_2 D(k+1)} | D(k) = t] d\mathbb{P}(D(k) < t) \\ &= \chi(s_2)\mathbb{E}[e^{-(s_1 + g(s_2))D(k)}], \end{aligned} \tag{5.5}$$

out of which the cross-moment of two consecutive downtimes $D(k)$ and $D(k+1)$ can be derived:

$$\begin{aligned} \mathbb{E}[D(k)D(k+1)] &= \frac{\partial}{\partial s_1}\frac{\partial}{\partial s_2}\chi(s_2)\mathbb{E}[e^{-(s_1+g(s_2))D(k)}]\Big|_{s_1=0, s_2=0} \\ &= -\chi'(0)\mathbb{E}[D(k)] + g'(0)\mathbb{E}[D(k)^2]. \end{aligned} \tag{5.6}$$

We obtain an expression for the bivariate Laplace-Stieltjes transform of $D(k)$ and $D(k+1)$ as $k \to \infty$ by combining (5.4) and (5.5):

$$\lim_{k\to\infty} \mathbb{E}[e^{-s_1 D(k) - s_2 D(k+1)}] = \chi(s_2) \prod_{j=0}^{\infty} \chi(g^{(j)}(s_1 + g(s_2))).$$

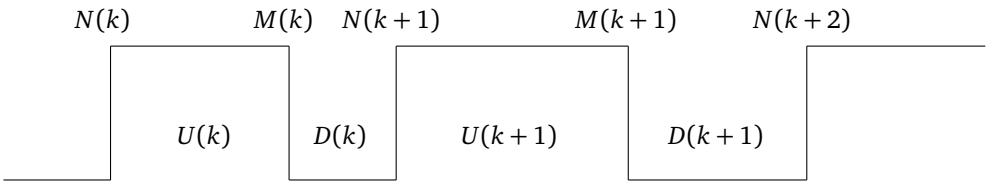

FIGURE 5.1: Two server up/down cycles.

Finally, the stability condition for the single-server model is given by

$$\rho = \lambda \mathbb{E}[B] < \frac{\mathbb{E}[U]}{\mathbb{E}[U] + \mathbb{E}[D]}.$$

5.3 Approximating the single-server model

We now focus on the queue length distribution of the single-server model with one-dependent vacation lengths. In particular, we derive an accurate approximation of the probability generating function of the queue length distribution. We later use this result to derive approximations for the extended machine repair model.

We first derive an approximation for the probability generating function of the distribution of N, the queue length at the beginning of an uptime, by studying the transient behaviour of the queue for two server up/down cycles. An observation length of one cycle would not suffice, since we explicitly need to take the dependence between consecutive downtimes (and thus dependence between cycle lengths) into account. Thus, we observe the system in its k-th uptime $U(k)$ as well as the following k-th downtime $D(k)$ and in the periods $U(k+1)$ and $D(k+1)$ thereafter. Referring to the queue length at the end of an uptime as M, let $N(k)$, $M(k)$, $N(k+1)$, $M(k+1)$ be the corresponding queue lengths; see Figure 5.1. For $k \to \infty$, we obviously have that

$$\mathbb{E}[z^{N(k)}] = \mathbb{E}[z^{N(k+2)}] = \mathbb{E}[z^N]. \tag{5.7}$$

In Section 5.3.1, we derive another expression of $\mathbb{E}[z^{N(k+2)}]$ in terms of $\mathbb{E}[z^{N(k)}]$, which holds approximately. We do this by deriving and connecting expressions for $\mathbb{E}[z^{M(k)}]$ in $\mathbb{E}[z^{N(k)}]$, $\mathbb{E}[z^{N(k+1)}]$ in $\mathbb{E}[z^{M(k)}]$ etc. We then approximate $\mathbb{E}[z^N]$ in Section 5.3.2 by combining the two expressions for $\mathbb{E}[z^{N(k+2)}]$ in $\mathbb{E}[z^{N(k)}]$ as $k \to \infty$. In Section 5.3.3, we use the results for the embedded times to obtain approximate expressions for $\mathbb{E}[z^M]$ and $\mathbb{E}[z^L]$, the probability generating functions corresponding to the queue length at the end of an uptime and at an arbitrary point in time, respectively. We conclude the analysis of the single-server model in Section 5.3.4 by illustrating the effects of dependence in downtimes. We believe that the analysis of a single-server queue with dependence between successive vacations is not only useful for studying the extended machine repair model, but is also of independent interest.

5.3.1 Behaviour of the queue length in two server up/down cycles

To obtain a relation between $\mathbb{E}[z^{N(k+2)}]$ and $\mathbb{E}[z^{N(k)}]$, we observe the way the queue length evolves in each of the periods $U(k)$, $D(k)$, $U(k+1)$ and $D(k+1)$. Connecting the

results then leads to an expression for $\mathbb{E}[z^{N(k+2)}]$ in terms of $\mathbb{E}[z^{N(k)}]$.

5.3.1.1 The queue length distribution during the first uptime

We first derive a relation between $\mathbb{E}[z^{M(k)}]$ and $\mathbb{E}[z^{N(k)}]$. During the first uptime $U(k)$, the server is accepting and processing customers. This means that the queue length in this period of time evolves similarly to the length of a regular M/G/1 queue during an exponential (σ) interval. This M/G/1 queue has the same customer arrival process and the same service time distribution, but does not have any service interruptions or server downtimes.

A relation between the probability generating functions of the queue length distribution at the beginning and the end of an exponentially distributed time interval in an M/G/1 queue can be obtained from the transition probabilities of the queue length between these two points in time. In [67, p. 246], these transition probabilities are derived as well as the resulting relation between the queue lengths at the beginning and the end of an exponentially distributed time interval. The relation between $M(k)$ and $N(k)$ in our context immediately follows:

$$\mathbb{E}[z^{M(k)}] = A(z)\mathbb{E}[z^{N(k)}] + K(z)\mathbb{E}[\widetilde{\mu}^{N(k)}(\sigma)], \tag{5.8}$$

where

$$A(z) = \frac{\sigma}{\sigma+\lambda(1-z)}\frac{z(1-\widetilde{B}(\sigma+\lambda(1-z)))}{z-\widetilde{B}(\sigma+\lambda(1-z))},$$
$$K(z) = -\frac{\sigma}{\sigma+\lambda(1-\widetilde{\mu}(\sigma))}\frac{(1-z)\widetilde{B}(\sigma+\lambda(1-z))}{z-\widetilde{B}(\sigma+\lambda(1-z))}$$

and where $\widetilde{\mu}(\sigma)$ is the Laplace-Stieltjes transform of (the distribution of) a busy period in the regular M/G/1 queue evaluated at σ. The value $\widetilde{\mu}(\sigma)$ is the unique root of the expression $z-\widetilde{B}(\sigma+\lambda(1-z))$ with $|\widetilde{\mu}(\sigma)|<1$ (for a proof of uniqueness, see [232, pp. 47–49]). Therefore, $\widetilde{\mu}(\sigma)$ is a pole of both $A(z)$ and $K(z)$, but these poles compensate each other. More specifically, by standard methods, we find the following result, which we will need in the sequel:

$$\begin{aligned}
&\lim_{z\to\widetilde{\mu}(\sigma)}\big(A(z)+K(z)\big)\\
&=\lim_{z\to\widetilde{\mu}(\sigma)}\Bigg(\frac{\sigma}{\sigma+\lambda(1-z)}\\
&\qquad+\bigg(\frac{\sigma}{\sigma+\lambda(1-z)}-\frac{\sigma}{\sigma+\lambda(1-\widetilde{\mu}(\sigma))}\bigg)\frac{(1-z)\widetilde{B}(\sigma+\lambda(1-z))}{z-\widetilde{B}(\sigma+\lambda(1-z))}\Bigg)\\
&=\frac{\sigma}{\sigma+\lambda(1-\widetilde{\mu}(\sigma))}+\frac{\lambda\widetilde{\mu}(\sigma)\sigma(1-\widetilde{\mu}(\sigma))}{\big(1+\lambda\widetilde{B}'(\sigma+\lambda(1-\widetilde{\mu}(\sigma)))\big)\big(\sigma+\lambda(1-\widetilde{\mu}(\sigma))\big)^2}.
\end{aligned} \tag{5.9}$$

5.3.1.2 The queue length distribution during the first downtime

During the first downtime $D(k)$, the server does not process any customers. Therefore, the queue length increases by the number of customer arrivals in this period. More specifically, the difference between $M(k)$ and $N(k+1)$ is exactly the number of Poisson arrivals during $D(k)$. It will prove convenient in later calculations to condition on the event

$D(k) = t$ for any $t \in \mathbb{R}_+$. Let $H(t)$ be Poisson (λt) distributed, i.e. the number of Poisson arrivals during $D(k) = t$. Observe that when downtimes exhibit autocorrelation, $H(t)$ and $M(k)$ are both correlated with the duration of the downtime preceding $D(k)$ and are thus interdependent too. To keep the analysis tractable, however, we assume that there is no interdependence between these two quantities; see also Section 5.3.1.6. As a result, we obtain the following approximate relation between $\mathbb{E}[z^{N(k+1)}|D(k) = t]$ and $\mathbb{E}[z^{M(k)}]$:

$$\begin{aligned}\mathbb{E}[z^{N(k+1)}|D(k) = t] &= \mathbb{E}[z^{M(k)+H(t)}] \\ &\approx \mathbb{E}[z^{M(k)}] \sum_{i=0}^{\infty} z^i e^{-\lambda t} \frac{(\lambda t)^i}{i!} = \mathbb{E}[z^{M(k)}] e^{-\lambda(1-z)t}. \end{aligned} \tag{5.10}$$

5.3.1.3 The queue length distribution during the second uptime

We now obtain a relation between $\mathbb{E}[z^{M(k+1)}|D(k) = t]$ and $\mathbb{E}[z^{N(k+1)}|D(k) = t]$. During the second uptime $U(k+1)$, the server is processing customers for an exponentially (σ) distributed amount of time, which means that the analysis is largely the same as the analysis of the queue length during the first uptime $U(k)$. The only difference stems from the fact that we now choose to condition on the event $D(k) = t$ in order to be able to concatenate all the results later on. Analogous to (5.8), we find

$$\begin{aligned}\mathbb{E}[z^{M(k+1)}|D(k) = t] &= A(z)\mathbb{E}[z^{N(k+1)}|D(k) = t] \\ &\quad + K(z)\mathbb{E}[\widetilde{\mu}^{N(k+1)}(\sigma)|D(k) = t]. \end{aligned} \tag{5.11}$$

5.3.1.4 The queue length distribution during the second downtime

To obtain a relation between $\mathbb{E}[z^{N(k+2)}|D(k) = t]$ and $\mathbb{E}[z^{M(k+1)}|D(k) = t]$, note that the server is not processing customers during the period $D(k+1)$, which again means that the difference between $M(k+1)$ and $N(k+2)$ is equal to the number of Poisson arrivals during the period $D(k+1)$. As described by (5.1), $D(k+1)$ is dependent on $D(k)$. Therefore, the previously introduced conditioning on the event $D(k) = t$ for $t \in \mathbb{R}_+$ is convenient at this point. By extending the analysis resulting in (5.10) to the second downtime conditional on the duration of the first downtime and implementing the dependence in (5.1), we obtain the following relation:

$$\begin{aligned}\mathbb{E}[z^{N(k+2)}|D(k) = t] &= \int_{u=0}^{\infty} \mathbb{E}[z^{M(k+1)+H(u)}|D(k) = t] \, d\mathbb{P}(D(k+1) < u|D(k) = t) \\ &= \int_{u=0}^{\infty} \mathbb{E}[z^{M(k+1)}|D(k) = t] e^{-\lambda(1-z)u} \, d\mathbb{P}(D(k+1) < u|D(k) = t) \\ &= \mathbb{E}[z^{M(k+1)}|D(k) = t]\mathbb{E}[e^{-\lambda(1-z)D(k+1)}|D(k) = t] \\ &= \mathbb{E}[z^{M(k+1)}|D(k) = t]\chi(\lambda(1-z))e^{-g(\lambda(1-z))t}, \end{aligned} \tag{5.12}$$

where $\mathbb{E}[z^{H(u)}] = e^{-\lambda(1-z)u}$ is the probability generating function corresponding to the number of Poisson arrivals during a time period with duration u.

5.3.1.5 Connecting all periods

Connecting the individual results corresponding to each of the periods, we now derive an expression for the conditional probability generating function $\mathbb{E}[z^{N(k+2)}|D(k)=t]$ in terms of $\mathbb{E}[z^{N(k)}]$. Keeping in mind (5.9) and the fact that $\widetilde{\mu}(\sigma)$ is a pole of $A(z)$ and $K(z)$, we note that for the substitution of $\mathbb{E}[\widetilde{\mu}^{M(k)}(\sigma)]$, the following important observation holds:

$$\begin{aligned}
&\lim_{z\to\widetilde{\mu}(\sigma)}\mathbb{E}[z^{M(k)}] = \lim_{z\to\widetilde{\mu}(\sigma)}\sum_{i=0}^{\infty}\big(A(z)z^i + K(z)\widetilde{\mu}^i(\sigma)\big)\mathbb{P}(N(k)=i)\\
&=\sum_{i=0}^{\infty}\Bigg(\lim_{z\to\widetilde{\mu}(\sigma)}\big(A(z)+K(z)\big)\widetilde{\mu}^i(\sigma) + \lim_{z\to\widetilde{\mu}(\sigma)}\big(A(z)(z^i-\widetilde{\mu}^i(\sigma))\big)\Bigg)\mathbb{P}(N(k)=i)\\
&=\sum_{i=0}^{\infty}\Bigg(\lim_{z\to\widetilde{\mu}(\sigma)}\big(A(z)+K(z)\big)\widetilde{\mu}^i(\sigma)\\
&\qquad + \frac{\sigma(1-\widetilde{\mu}(\sigma))}{(\sigma+\lambda(1-\widetilde{\mu}(\sigma)))(1+\lambda\widetilde{B}'(\sigma+\lambda(1-\widetilde{\mu}(\sigma))))}i\widetilde{\mu}^i(\sigma)\Bigg)\mathbb{P}(N(k)=i)\\
&=\Bigg(\lim_{z\to\widetilde{\mu}(\sigma)}\big(A(z)+K(z)\big)\Bigg)\mathbb{E}[\widetilde{\mu}^{N(k)}(\sigma)]\\
&\quad + \frac{\sigma(1-\widetilde{\mu}(\sigma))}{(\sigma+\lambda(1-\widetilde{\mu}(\sigma)))(1+\lambda\widetilde{B}'(\sigma+\lambda(1-\widetilde{\mu}(\sigma))))}\mathbb{E}[N(k)\widetilde{\mu}^{N(k)}(\sigma)].
\end{aligned}$$

As a result, an extra term containing the expression $\mathbb{E}[N(k)\widetilde{\mu}^{N(k)}(\sigma)]$ arises in the expression for $\mathbb{E}[z^{N(k+2)}|D(k)=t]$. More specifically, by combining (5.8), (5.10), (5.11) and (5.12), we obtain

$$\begin{aligned}
&\mathbb{E}[z^{N(k+2)}|D(k)=t]\\
&\approx \chi(\lambda(1-z))A^2(z)e^{-(\lambda(1-z)+g(\lambda(1-z)))t}\mathbb{E}[z^{N(k)}]\\
&\quad + \chi(\lambda(1-z))K(z)\Bigg(A(z)e^{-(g(\lambda(1-z))+\lambda(1-z))t}\\
&\qquad\qquad + \lim_{p\to\widetilde{\mu}(\sigma)}\big(A(p)+K(p)\big)e^{-(g(\lambda(1-z))+\lambda(1-\widetilde{\mu}(\sigma)))t}\Bigg)\mathbb{E}[\widetilde{\mu}^{N(k)}(\sigma)]\\
&\quad + \chi(\lambda(1-z))K(z)e^{-(g(\lambda(1-z))+\lambda(1-\widetilde{\mu}(\sigma)))t}\\
&\qquad \times \frac{\sigma(1-\widetilde{\mu}(\sigma))}{(\sigma+\lambda(1-\widetilde{\mu}(\sigma)))(1+\lambda\widetilde{B}'(\sigma+\lambda(1-\widetilde{\mu}(\sigma))))}\mathbb{E}[N(k)\widetilde{\mu}^{N(k)}(\sigma)]. \qquad (5.13)
\end{aligned}$$

In the course of the previous calculations, we conditioned on the event $D(k)=t$. In the expression for $\mathbb{E}[z^{N(k+2)}|D(k)=t]$, we see that the value t is only found in the form $e^{-st}(s\geq 0)$, meaning that unconditioning leads to expressions in terms of the Laplace-Stieltjes transform $\widetilde{D}(\cdot)$:

$$\begin{aligned}
\mathbb{E}[z^{N(k+2)}] &\approx \int_{t=0}^{\infty}\mathbb{E}[z^{N(k+2)}|D(k)=t]d\mathbb{P}(D(k)<t)\\
&= E(z)\mathbb{E}[z^{N(k)}] + F(z)\mathbb{E}[\widetilde{\mu}^{N(k)}(\sigma)] + G(z)\mathbb{E}[N(k)\widetilde{\mu}^{N(k)}(\sigma)], \qquad (5.14)
\end{aligned}$$

where

$$
\begin{aligned}
E(z) &= \chi(\lambda(1-z))A^2(z)\widetilde{D}(\lambda(1-z)+g(\lambda(1-z))),\\
F(z) &= \chi(\lambda(1-z))K(z)\Big(A(z)\widetilde{D}(\lambda(1-z)+g(\lambda(1-z)))\\
&\qquad + \widetilde{D}(\lambda(1-\widetilde{\mu}(\sigma))+g(\lambda(1-z)))\lim_{p\to\widetilde{\mu}(\sigma)}(A(p)+K(p))\Big),\\
G(z) &= \chi(\lambda(1-z))K(z)\widetilde{D}(\lambda(1-\widetilde{\mu}(\sigma))+g(\lambda(1-z)))\\
&\qquad \times \frac{\sigma(1-\widetilde{\mu}(\sigma))}{(\sigma+\lambda(1-\widetilde{\mu}(\sigma)))(1+\lambda\widetilde{B}(\sigma+\lambda(1-\widetilde{\mu}(\sigma))))}.
\end{aligned}
\tag{5.15}
$$

This expression gives a relation between $\mathbb{E}[z^{N(k+2)}]$ and $\mathbb{E}[z^{N(k)}]$.

5.3.1.6 A note on the approximation assumptions made

Except when the downtimes are completely independent (i.e. $g(s) = 0$), the relation between $\mathbb{E}[z^{N(k+2)}]$ and $\mathbb{E}[z^{N(k)}]$ given in (5.14) only holds approximately as opposed to exactly. This is the case, since (5.14) is among other expressions based on (5.10). The latter expression is approximate of nature, since we assumed $D(k)$, the k-th downtime, and $M(k)$, the queue length at the end of the k-th uptime, to be independent. In reality, this is not the case, since $D(k)$ and $M(k)$ are both correlated with $D(k-1)$, the period of downtime preceding $D(k)$. Thus, these two quantities are mutually correlated too.

When we drop the approximation assumption of independence between $D(k)$ and $M(k)$, however, the analysis becomes considerably harder. To account for the dependence, one would have to condition throughout on the event $D(k-1) = s$ instead of the event $D(k) = t$. Equivalent expressions to (5.8), (5.10), (5.11) and (5.12) can still be obtained in the same fashion as before:

$$
\begin{aligned}
\mathbb{E}[z^{M(k)} \mid D(k-1)=s] &= A(z)\mathbb{E}[z^{N(k)} \mid D(k-1)=s] + K(z)\mathbb{E}[\widetilde{\mu}^{N(k)}(\sigma) \mid D(k-1)=s],\\
\mathbb{E}[z^{N(k+1)} \mid D(k-1)=s] &= \mathbb{E}[z^{M(k)} \mid D(k-1)=s]\chi(\lambda(1-z))e^{-g(\lambda(1-z))s},\\
\mathbb{E}[z^{M(k+1)} \mid D(k-1)=s] &= A(z)\mathbb{E}[z^{N(k+1)} \mid D(k-1)=s]\\
&\qquad + K(z)\mathbb{E}[\widetilde{\mu}^{N(k+1)}(\sigma) \mid D(k-1)=s]
\end{aligned}
$$

and

$$
\begin{aligned}
\mathbb{E}[z^{N(k+2)} \mid D(k-1)=s] &= \mathbb{E}[z^{M(k+1)} \mid D(k-1)=s]\\
&\qquad \times \chi(\lambda(1-z))\chi(g(\lambda(1-z)))e^{-g(g(\lambda(1-z)))s}.
\end{aligned}
$$

Concatenating these results leads to a relation of the following form:

$$
\begin{aligned}
\mathbb{E}[z^{N(k+2)} \mid D(k-1)=s] &= E(s,z)\mathbb{E}[z^{N(k)} \mid D(k-1)=s]\\
&\qquad + F(s,z)\mathbb{E}[\mu^{N(k)}(\sigma) \mid D(k-1)=s]\\
&\qquad + G(s,z)\mathbb{E}[N(k)\mu^{N(k)}(\sigma) \mid D(k-1)=s].
\end{aligned}
$$

Extracting a relation between $\mathbb{E}[z^{N(k+2)}]$ and $\mathbb{E}[z^{N(k)}]$ from this expression is not straightforward, as s appears in both the coefficients and the expectations of the right-hand side of this equation.

Observe, however, that the assumed independence between $D(k)$ and $M(k)$ is the only source of approximation error that we introduce in the entire Section 5.3. Moreover, numerical results suggest that this approximation error is generally very small. In fact, the correlation between the downtimes needs to be very strong in order for the approximation error to be noticeable. As the main goal of this chapter is to derive an approximation for the (marginal) queue length distribution of the first-layer queues in the extended machine repair model, we choose not to extensively discuss these numerical experiments. Instead, in Section 5.5, we will present and discuss numerical results for the approximation that we eventually obtain for the extended machine repair model. It will turn out that the accuracy of this final approximation is very good, while this approximation actually includes another source of error that we introduce in Section 5.4 to connect the single server model and the extended machine repair model.

5.3.2 Queue length at the beginning of an arbitrary uptime

We now obtain an expression for $\mathbb{E}[z^N] = \lim_{k\to\infty} \mathbb{E}[z^{N(k)}]$. Combining (5.7) and (5.14), we find

$$\mathbb{E}[z^N] \approx \frac{F(z)\mathbb{E}[\widetilde{\mu}^N(\sigma)] + G(z)\mathbb{E}[N\widetilde{\mu}^N(\sigma)]}{1-E(z)} \tag{5.16}$$

with $E(z), F(z)$ and $G(z)$ as given in (5.15). Observe that this expression has two unknown constants $\mathbb{E}[\widetilde{\mu}^N(\sigma)]$ and $\mathbb{E}[N\widetilde{\mu}^N(\sigma)]$. We show that these constants can be obtained approximately as the solution of a system of two linear equations. These two equations lead to a unique solution for $\mathbb{E}[\widetilde{\mu}^N(\sigma)]$ and $\mathbb{E}[N\widetilde{\mu}^N(\sigma)]$. We derive them below. Expressions for the constants immediately follow.

The case $z = 1$ Since the left-hand side of (5.16) evaluates to one for $z = 1$ and $F(1) = G(1) = 1 - E(1) = 0$, we have for the right-hand side that

$$\lim_{z\to 1} \frac{F(z)\mathbb{E}[\widetilde{\mu}^N(\sigma)] + G(z)\mathbb{E}[N\widetilde{\mu}^N(\sigma)]}{1-E(z)} = -\frac{F'(1)\mathbb{E}[\widetilde{\mu}^N(\sigma)] + G'(1)\mathbb{E}[N\widetilde{\mu}^N(\sigma)]}{E'(1)} \approx 1$$

by l'Hôpital's rule. Since $E(z)$, $F(z)$ and $G(z)$ are each differentiable at $z = 1$, this results in the first linear equation in the two unknowns $\mathbb{E}[\widetilde{\mu}^N(\sigma)]$ and $\mathbb{E}[N\widetilde{\mu}^N(\sigma)]$.

The case $z = \phi$ The denominator $1-E(z)$ of (5.16) has a root $z = \phi$ between zero and $\widetilde{\mu}(\sigma) < 1$. More specifically, the following lemma holds.

LEMMA 5.3.1. *The denominator* $1-E(z)$ *has exactly one root on the real line in the domain* $(0, \widetilde{\mu}(\sigma))$.

PROOF. See Appendix 5.A. □

Let ϕ be the unique root mentioned in Lemma 5.3.1. Since $\mathbb{E}[z^N]$ is analytic in z for $|z| \leq 1$ and thus cannot evaluate to $\pm\infty$ for $0 < z < \widetilde{\mu}(\sigma)$, we have that this root should also be a root for the numerator. Hence, we have that $F(\phi)\mathbb{E}[\widetilde{\mu}^N(\sigma)] + G(\phi)\mathbb{E}[N\widetilde{\mu}^N(\sigma)] = 0$.

Combining (5.16) with the cases $z = 1$ and $z = \phi$, we conclude that the probability generating function corresponding to the queue length at the beginning of an arbitrary uptime is well approximated by

$$\mathbb{E}[z^N] \approx \frac{F(z)\mathbb{E}[\widetilde{\mu}^N(\sigma)] + G(z)\mathbb{E}[N\widetilde{\mu}^N(\sigma)]}{1-E(z)}, \tag{5.17}$$

where

$$\mathbb{E}[\widetilde{\mu}^N(\sigma)] \approx \frac{E'(1)G(\phi)}{F(\phi)G'(1)-F'(1)G(\phi)} \text{ and } \mathbb{E}[N\widetilde{\mu}^N(\sigma)] \approx \frac{E'(1)F(\phi)}{F'(1)G(\phi)-F(\phi)G'(1)}.$$

5.3.3 Queue length at an arbitrary point in time

The main goal of this section is to study the probability generating function of the queue length distribution at an arbitrary point in time. To obtain an approximate expression for this function, we expand the results of the previous section. An expression for the probability generating function $\mathbb{E}[z^M]$ of the queue length at the start of an arbitrary downtime is easily derived from the probability generating function $\mathbb{E}[z^N]$ of the queue length at the start of an arbitrary uptime. We then derive an expression for the probability generating functions corresponding to the queue length observed at an arbitrary point within an uptime and the queue length observed at an arbitrary point within a downtime, respectively. As a result, we finally obtain an approximate expression for $\mathbb{E}[z^L]$, the probability generating function corresponding to the queue length at an arbitrary point in time.

5.3.3.1 Observing the queue length during an arbitrary uptime

To obtain the distribution of the queue length at an arbitrary point during an arbitrary uptime, we first derive an expression for $\mathbb{E}[z^M]$. By letting $k \to \infty$ in (5.10) after the necessary integration to remove the condition $D(k) = t$, we obtain

$$\mathbb{E}[z^N] \approx \lim_{k\to\infty} \int_{t=0}^{\infty} \mathbb{E}[z^{M(k)}] e^{-\lambda(1-z)t} d\mathbb{P}(D(k) < t) = \mathbb{E}[z^M]\widetilde{D}(\lambda(1-z)). \tag{5.18}$$

Next, we make use of the following lemma.

LEMMA 5.3.2. *The probability generating function corresponding to the queue length at an arbitrary point in an uptime satisfies*

$$\mathbb{E}[z^L | \textit{server up}] = \mathbb{E}[z^M].$$

PROOF. Let $V(t)$ be the number of vacation initiations of the server in $(0, t]$. Note that $V(t)$ is a doubly stochastic process, where during the uptime of a server, initiations of vacations occur according to a Poisson process with rate σ, whereas they obviously occur with rate zero when the server is already on a vacation. The conditional PASTA property (cf. [257]) applied to $V(t)$ implies that the queue length distribution at the start of a vacation equals the queue length distribution at an arbitrary point in time during an uptime. □

Combining (5.18) with Lemma 5.3.2 now yields that

$$\mathbb{E}[z^L|\textit{server up}] \approx \frac{\mathbb{E}[z^N]}{\widetilde{D}(\lambda(1-z))}, \tag{5.19}$$

where $\mathbb{E}[z^N]$ is (approximately) given by (5.17).

REMARK 5.3.1. An expression for $\mathbb{E}[z^M]$ into $\mathbb{E}[z^N]$ is also readily given by (5.8) when taking the limit $k \to \infty$. Using Lemma 5.3.2, this expression leads to an alternative expression for the probability generating function of the queue length distribution when observed during a downtime:

$$\mathbb{E}[z^L|\textit{server up}] = A(z)\mathbb{E}[z^N] + K(z)\mathbb{E}[\widetilde{\mu}^N(\sigma)],$$

where $A(z)$, $K(z)$ and $\widetilde{\mu}(\sigma)$ are defined as before.

5.3.3.2 Observing the queue length during a downtime

At an arbitrary point in time during a downtime, the number of customers in the system can be decomposed into the number of customers already waiting at the end of the previous uptime M and the number of customers who arrived during the elapsed time D^{past} since the start of the current downtime, which we denote by $H(D^{past})$. Note that M and $H(D^{past})$ are not independent. A large value of M may imply that the previous downtime has been very long. Due to the positive correlation between the downtimes as assumed in both models, this would in turn imply that the current downtime is probably longer than usual as well. The duration of the current downtime and its past part D^{past} are obviously dependent, which results in the fact that M and $H(D^{past})$ are dependent. Using the notation illustrated in Figure 5.1, we obtain

$$\begin{aligned}\mathbb{E}[z^L|\textit{server down}] &= \mathbb{E}[z^{M+H(D^{past})}] \\ &= \lim_{k\to\infty} \int_0^\infty \mathbb{E}[z^{M(k+1)}|D(k)=t]\mathbb{E}[z^{H(D^{past}(k+1))}|D(k)=t]d\mathbb{P}(D(k)<t). \end{aligned} \tag{5.20}$$

From the intermediate calculations leading to (5.14) (or by simply combining (5.12) and (5.13)), we have that

$$\lim_{k\to\infty} \mathbb{E}[z^{M(k+1)}|D(k)=t] \approx \sum_{i=1}^{2} q_i(z)e^{-r_i(z)t}, \tag{5.21}$$

where

$$q_1(z) = A(z)(A(z)\mathbb{E}[z^N] + K(z)\mathbb{E}[\widetilde{\mu}^N(\sigma)]), \tag{5.22}$$

$$\begin{aligned} q_2(z) = K(z)\Bigg(\Bigg(\lim_{p\to\widetilde{\mu}(\sigma)}(A(p)+K(p))\Bigg)\mathbb{E}[\widetilde{\mu}^N(\sigma)] \\ + \frac{\sigma(1-\widetilde{\mu}(\sigma))}{(\sigma+\lambda(1-\widetilde{\mu}(\sigma)))(1+\lambda\widetilde{B}'(\sigma+\lambda(1-\widetilde{\mu}(\sigma))))}\mathbb{E}[N\widetilde{\mu}^N(\sigma)]\Bigg), \end{aligned} \tag{5.23}$$

$$r_1(z) = \lambda(1-z) \textit{ and } r_2(z) = \lambda(1-\widetilde{\mu}(\sigma)). \tag{5.24}$$

Furthermore, from (5.1), we obtain

$$\begin{aligned}\mathbb{E}[z^{H(D^{past}(k+1))}|D(k)=t] &= \mathbb{E}[e^{-\lambda(1-z)D^{past}(k+1)}|D(k)=t] \\ &= \frac{1-\mathbb{E}[e^{-\lambda(1-z)D(k+1)}|D(k)=t]}{\lambda(1-z)\mathbb{E}[D(k+1)|D(k)=t]} \\ &= \frac{1-\chi(\lambda(1-z))e^{-g(\lambda(1-z))t}}{\lambda(1-z)(g'(0)t-\chi'(0))}. \end{aligned} \tag{5.25}$$

Combining (5.21)–(5.25), we have that the evaluation of (5.20) involves the computation of a linear combination of integrals with the form

$$\int_{t=0}^{\infty} \frac{e^{-at}}{bt+c} d\mathbb{P}(D<t) = \int_{t=0}^{\infty}\int_{u=0}^{\infty} e^{-(at+(bt+c)u)} du\, d\mathbb{P}(D<t).$$

By interchanging the integrals, this expression reduces to

$$\kappa_{a,b}(c) = \int_0^{\infty} e^{-cu}\widetilde{D}(a+bu)du,$$

i.e. the Laplace transform of the function $\widetilde{D}(a+bu)$. Combining all of the results, we have that the probability generating function of the queue length distribution at an arbitrary point in a downtime is approximately given by

$$\mathbb{E}[z^L|\textit{server down}] \tag{5.26}$$

$$\begin{aligned}&\approx \int_{t=0}^{\infty}\left(\sum_{i=1}^{2} q_i(z)e^{-r_i(z)t}\frac{1-\chi(\lambda(1-z))e^{-g(\lambda(1-z))t}}{\lambda(1-z)(g'(0)t-\chi'(0))}\right)d\mathbb{P}(D<t) \\ &= \frac{1}{\lambda(1-z)}\sum_{i=1}^{2} q_i(z)\Big(\kappa_{r_i(z),g'(0)}(-\chi'(0)) \\ &\qquad - \chi(\lambda(1-z))\kappa_{r_i(z)+g(\lambda(1-z)),g'(0)}(-\chi'(0))\Big),\end{aligned} \tag{5.27}$$

where $\kappa_{a,b}(c) = \int_0^{\infty} e^{-cu}\widetilde{D}(a+bu)du$. Note that in case $\widetilde{D}(\cdot)$ is not explicitly known by inspecting (5.2), one can still evaluate $\kappa_{a,b}(c)$ up to arbitrary precision by truncating the infinite product in (5.4).

5.3.3.3 Deriving the general queue length distribution

From the results derived for the queue length conditioned on the different states of the server, an approximation for the unconditional queue length distribution of the single-server model can be derived, which results in the following statement.

APPROXIMATION 5.3.3. *The probability generating function of the queue length distribution in the single-server model with one-dependent downtimes is given by*

$$\mathbb{E}[z^L] = p_{up}\mathbb{E}[z^L|\textit{server up}] + p_{down}\mathbb{E}[z^L|\textit{server down}], \tag{5.28}$$

where

$$p_{up} = \frac{\mathbb{E}[U]}{\mathbb{E}[U]+\mathbb{E}[D]} = \frac{1}{1+\sigma\mathbb{E}[D]} \textit{ and } p_{down} = \frac{\mathbb{E}[D]}{\mathbb{E}[U]+\mathbb{E}[D]} = \frac{\sigma\mathbb{E}[D]}{1+\sigma\mathbb{E}[D]}, \tag{5.29}$$

and approximate expressions for $\mathbb{E}[z^L|$*server up*$]$ *and* $\mathbb{E}[z^L|$*server down*$]$ *are given by* (5.19) *and* (5.27)*, respectively.*

The weights p_{up} and p_{down} are the probabilities that one finds the server up and down, respectively, when observing the system at a random point in time in steady state. These probabilities are derived through the straightforward application of Palm theory (cf. [23, 220]) and involve the computation of $\mathbb{E}[U]$ and $\mathbb{E}[D]$. The former is determined by the fact that U is exponentially (σ) distributed, and the latter follows from (5.3). In Section 5.4, we will use this approximation obtained for the (probability generating function of the) queue length distribution of the single-server model as a basis for the derivation of an approximation for the marginal queue length distribution of a first-layer queue in the extended machine repair model.

REMARK 5.3.2. Observe that the evaluation of Approximation 5.3.3 involves the evaluation of several values of the Laplace-Stieltjes transform $\widetilde{D}(\cdot)$. Whenever the Laplace-Stieltjes transform cannot be derived by solving the functional equation (5.2), computing the values of $\widetilde{D}(\cdot)$ is not possible in an exact fashion. However, we can use the infinite-product representation (5.4) to derive these values up to arbitrary precision. This product converges fast and therefore truncation leads to an arbitrarily accurate approximation. The numerical experiments in Section 5.5 also confirm this fast convergence.

REMARK 5.3.3. The analysis of the single-server queue as presented in this section can be extended to dependence forms that are different from (5.1). For example, for Markov-modulated dependencies the same strategy can be used to obtain approximate expressions for the queue length distributions. Slight adaptations have to be made in the computations, starting with the conditional Laplace-Stieltjes transform in (5.12).

5.3.4 A note on the impact of dependence

Now that we have obtained an accurate approximation of the probability generating function of the queue length distribution, we numerically study the influence of the downtime dependence on the queue length distribution. We will show that the level of dependence between the downtimes influences the queue length distribution considerably. Observe an instance of the single-server model where $\lambda = 3$, the service time B is exponentially distributed with rate 5 and the uptime U of the server is exponentially distributed with rate $1/3$. In this particular example, the downtime of the server consists of multiple exponential phases. The number of phases of which a downtime $D(k+1)$ consists depends on the previous downtime $D(k)$:

$$D(k+1) \stackrel{d}{=} C_1 + \cdots + C_{J(D(k))+1}, \tag{5.30}$$

where the C_i, which represent the phases, are independent and exponentially (δ) distributed, $\delta > 1$, and $J(D(k))$ is Poisson distributed with parameter $D(k)$. This implies that

$$\begin{aligned}\mathbb{E}[e^{-sD(k+1)}|D(k)=t] &= \sum_{j=0}^{\infty} \mathbb{E}[e^{-s(C_1+\sum_{i=2}^{j+1} C_i)}]e^{-t}\frac{t^j}{j!} = \mathbb{E}[e^{-sC_1}]\sum_{j=0}^{\infty}\mathbb{E}[e^{-sC_1}]^j e^{-t}\frac{t^j}{j!} \\ &= \mathbb{E}[e^{-sC_1}]e^{-(1-\mathbb{E}[e^{-sC_1}])t}.\end{aligned}$$

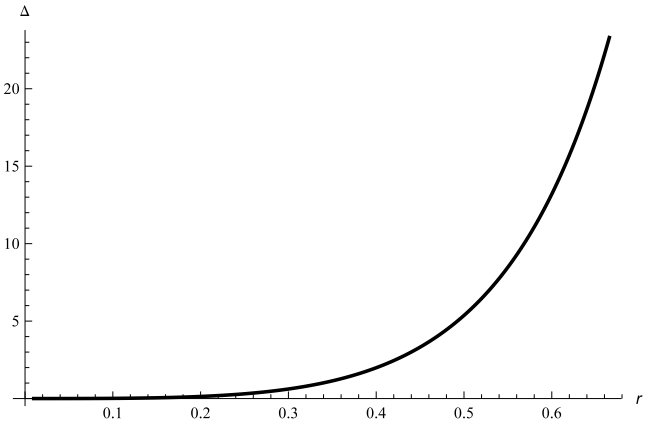

FIGURE 5.2: The percentual relative difference Δ in $\mathbb{E}[L]$ between the dependent and the independent model for various values of the correlation coefficient r.

Therefore, we have that $\chi(s) = \mathbb{E}[e^{-sC_1}] = \frac{\delta}{\delta+s}$ and $g(s) = 1 - \mathbb{E}[e^{-sC_1}] = \frac{s}{\delta+s}$. The stationary downtime is exponentially $(\delta - 1)$ distributed, since (5.2) is satisfied for its Laplace-Stieltjes transform $\widetilde{D}(s) = \frac{\delta-1}{\delta-1+s}$. Observe that the stationary downtime distribution only exists for $\delta > 1$.

We compare the model above with its 'independent counterpart', namely a single-server queue with the same interarrival, service, uptime and stationary downtime distributions as before, but with mutually independent downtimes. The independent downtimes also fit in the dependence structure of (5.1) by simply setting $g(s) = 0$ for all s. Since the stationary downtime distribution is exponentially $(\delta-1)$ distributed, we trivially have for the independent model that $\chi(s) = \widetilde{D}(s) = \frac{\delta-1}{\delta-1+s}$ and $g(s) = 0$.

To see the effect of the dependencies, we compare $\mathbb{E}[L^{dep}]$, the (approximated) expected queue length in the dependent model, with $\mathbb{E}[L^{indep}]$, the expected queue length of the independent model. These values are obtained by evaluating the derivative of (5.28) at $z = 1$. We compute the percentual relative difference of both quantities, i.e.

$$\Delta = 100\% \times \frac{\mathbb{E}[L^{dep}] - \mathbb{E}[L^{indep}]}{\mathbb{E}[L^{indep}]},$$

for varying values of δ such that the load of the system varies between 0.6 and 1. For the dependent model, the value of δ determines the correlation coefficient between two consecutive downtimes in steady state, which we denote by r. More specifically, by the definition of the correlation coefficient, we have that

$$r = \frac{\lim_{k\to\infty} \mathbb{E}[D(k)D(k+1)] - (\mathbb{E}[D])^2}{\mathbb{E}[D^2] - (\mathbb{E}[D])^2}. \tag{5.31}$$

This expression can be given in terms of δ by using (5.3) and (5.6). For the independent model, the correlation coefficient between the downtimes obviously equals zero at all times. Figure 5.2 shows the value of Δ as a function of the correlation r as observed in the dependent model. We see in this figure that Δ equals zero for $r = 0$, while Δ grows

as high as 25% for increasing r. This figure shows that the correlation in the downtimes can have a large impact on the queue length and should thus not be ignored.

5.4 Approximating the extended machine repair model

In this section, we use Approximation 5.3.3 to derive an approximation for the marginal queue length distributions in the extended machine repair model. We do this by connecting the single-server model with the extended machine-repair model, which requires another approximation step. To connect these models, we observe that arrival streams, service times and uptimes are equivalent for both models. To describe the downtime distribution and the dependence of the downtimes in the extended machine repair model in terms of the parameters of the single-server model as well as possible, we need to obtain suitable choices for the functions $\chi(s)$ and $g(s)$, which are used in (5.1). When investigating L_i, the queue length of Q_i, we choose suitable functions $\chi_i(s)$ and $g_i(s)$ that are specific to M_i, $i = 1, 2$. The resulting explicit downtime structure matches the downtime distribution and downtime dependence of the downtimes of M_i in the extended machine repair model closely, but does not model it exactly. Thus, apart from the approximation assumption discussed in Section 5.3.1.6, this forms another source of approximation error. However, numerical results in Section 5.5 will show the final approximation to be very accurate.

Evidently, the accuracy of the final approximation depends among other things on the quality of the choices for $\chi_i(s)$ and $g_i(s)$. Therefore, we first focus on how to choose these functions appropriately. For this purpose, we compute in Section 5.4.1 the first two moments and the correlation coefficient of consecutive downtimes in the extended machine repair model. Based on these numbers, Section 5.4.2 derives suitable choices for the functions $\chi_i(s)$ and $g_i(s)$ such that they match the situation in the extended machine repair model as well as possible. After these preliminary steps, we combine these results with those of the previous section to obtain an approximation for the (probability generating function of the) distribution of L_i, which is one of the main results of this chapter, in Section 5.4.3. This approximation is applicable for the extended machine repair model with two machines and a single repairman. However, the approach we follow remains valid for more general models. We discuss this in Section 5.4.4.

5.4.1 Moments and the correlation coefficient of the downtimes

In this section, we focus on exponential repair times. The analysis can be extended to phase-type repair times, but at the cost of more cumbersome expressions that offer little additional insight. We derive the first two moments of the stationary downtime distribution of machine M_1 as well as the correlation coefficient between two consecutive downtimes $D_1(k)$ and $D_1(k+1)$ in steady state (i.e. for $k \to \infty$). We do this by studying the two-dimensional Laplace-Stieltjes transform $\mathbb{E}[e^{-s_1 D_1(k) - s_2 D_1(k+1)}]$. Evidently, a downtime $D_1(k)$ can be decomposed into a waiting time $W_1(k)$ and a repair time $R_1(k)$. The waiting time $W_1(k)$ is either zero when M_2 is operational at the time of breakdown of M_1 or amounts to an exponentially (ν_2) distributed residual of the repair time of M_2 otherwise.

Assume that the repairman repairs M_1 and M_2 at rate ν_1 and ν_2, respectively. As noted before, machines interfere with each other in the extended machine repair model through their downtimes. More specifically, we have that a lengthy repair time of M_1 may increase

the waiting time in the next downtime of M_2. At the same time, a lengthy downtime for M_2 may have an increasing influence on the next waiting time of M_1. Therefore, $R_1(k)$ and $W_1(k+1)$ are positively correlated. Thus, the two-dimensional Laplace-Stieltjes transform of two consecutive downtimes may be written as

$$\begin{aligned}&\mathbb{E}[e^{-s_1 D_1(k)-s_2 D_1(k+1)}]\\&= \mathbb{E}[e^{-s_1 W_1(k)}]\mathbb{E}[e^{-s_2 R_1(k+1)}]\int_0^\infty e^{-s_1 y}\mathbb{E}[e^{-s_2 W_1(k+1)}|R_1(k)=y]\,\nu_1 e^{-\nu_1 y}\,dy. \qquad (5.32)\end{aligned}$$

Since $R_1(k+1)$ is exponentially (ν_1) distributed (i.e. $\mathbb{E}[e^{-s_2 R_1(k+1)}] = \frac{\nu_1}{\nu_1+s_2}$), only the transforms $\mathbb{E}[e^{-s_1 W_1(k)}]$ and $\mathbb{E}[e^{-s_2 W_1(k+1)}|R_1(k)=y]$ remain to be computed.

First, we derive $\mathbb{E}[e^{-s_1 W_1(k)}]$. Just before M_1 breaks down, either M_2 is up and running, or M_2 is in repair. The probability of either event happening is derived by studying the embedded discrete-time Markov chain of the machine states at epochs where any machine breaks down, starts being repaired or ends a repair period. Let $\boldsymbol{X}_n = (X_{1,n}, X_{2,n})$ denote the state of the machines after the n-th transition. As before, we represent the state of M_i being up, waiting for repair or being in repair after the n-th transition by $X_{i,n} = U$, $X_{i,n} = R$ or $X_{i,n} = W$, respectively. Observe that $\{\boldsymbol{X}_n, n \geq 0\}$ is a discrete-time Markov chain on the state space $\mathscr{S}$ as given in Section 2.2. It naturally follows that the non-zero transition probabilities $p_{i,j}$ from state $i \in \mathscr{S}$ to state $j \in \mathscr{S}$ are given by $p_{(U,U),(R,U)} = 1 - p_{(U,U),(U,R)} = \frac{\sigma_1}{\sigma_1+\sigma_2}$, $p_{(U,R),(U,U)} = 1 - p_{(U,R),(W,R)} = \frac{\nu_2}{\sigma_1+\nu_2}$, $p_{(R,U),(U,U)} = 1 - p_{(R,U),(R,W)} = \frac{\nu_1}{\nu_1+\sigma_2}$ and $p_{(W,R),(R,U)} = p_{(R,W),(U,R)} = 1$. The discrete-time Markov chain is irreducible and aperiodic, hence a unique limiting distribution π' for $\{\boldsymbol{X}_n, n \geq 0\}$ exists and can be derived. Given this distribution, the probability of an arbitrary transition being an event where M_1 breaks down equals $\pi'_{(U,U)}p_{(U,U),(R,U)} + \pi'_{(U,R)}p_{(U,R),(W,R)}$. The probability z_{up} (z_{down}) of M_2 working (being in repair), given that M_1 breaks down next transition, is thus given by

$$\begin{aligned}z_{up} &= \frac{\pi'_{(U,U)}p_{(U,U),(R,U)}}{\pi'_{(U,U)}p_{(U,U),(R,U)} + \pi'_{(U,R)}p_{(U,R),(W,R)}} = \frac{\sigma_1\nu_1 + (\sigma_2+\nu_1)\nu_2}{(\sigma_2+\nu_1)(\sigma_1+\sigma_2+\nu_2)},\\ z_{down} &= \frac{\pi'_{(U,R)}p_{(U,R),(W,R)}}{\pi'_{(U,U)}p_{(U,U),(R,U)} + \pi'_{(U,R)}p_{(U,R),(W,R)}} = \frac{\sigma_2(\sigma_1+\sigma_2+\nu_1)}{(\sigma_2+\nu_1)(\sigma_1+\sigma_2+\nu_2)}.\end{aligned}$$

Hence, M_1 has to wait with probability z_{down}, whereas it does not with probability z_{up}. Therefore, we have that

$$\begin{aligned}\mathbb{E}[e^{-s_1 W_1(k)}] &= z_{up} + z_{down}\frac{\nu_2}{\nu_2+s_1}\\ &= \frac{s_1\sigma_1\nu_1 + s_1(\sigma_2+\nu_1)\nu_2 + (\sigma_2+\nu_1)\nu_2(\sigma_1+\sigma_2+\nu_2)}{(\sigma_2+\nu_1)(s_1+\nu_2)(\sigma_1+\sigma_2+\nu_2)}.\end{aligned}$$

For $\mathbb{E}[e^{-s_2 W_1(k+1)}|R_1(k)=y]$, we first conclude that at the moment M_1 is taken into repair for y time units, M_2 must be working. After these y time units, we have a probability $e^{-\nu_2 y}$ of M_2 having broken down in the meantime, whereas it is still functioning with probability $1-e^{-\nu_2 y}$. Given the former event that M_2 is still working at the end of $R_1(k)$, there is a probability u that M_2 is in repair when M_1 breaks down again, i.e. at the start of $W_1(k+1)$. Due to the memoryless property of the exponential distribution, this probability

u is easily determined by the fixed point equation

$$u = \frac{\sigma_2}{\sigma_1+\sigma_2}\left(\frac{\sigma_1}{\sigma_1+\nu_2} + \frac{\nu_2}{\sigma_1+\nu_2}u\right),$$

which leads to

$$u = \frac{\sigma_2}{\sigma_1+\sigma_2+\nu_2}.$$

This allows us to determine the probability w that M_2 is in repair at the start of $W_1(k+1)$, given that M_2 was waiting for repair at the end of $R_1(k)$:

$$w = \frac{\sigma_1}{\sigma_1+\nu_2} + \frac{\nu_2}{\sigma_1+\nu_2}u = \frac{\sigma_1+\sigma_2}{\sigma_1+\sigma_2+\nu_2}.$$

Taking these probabilities together, we have that $W_1(k+1)$ is exponentially (ν_2) distributed with probability $e^{-\nu_2 y}u + (1-e^{-\nu_2 y})w$ and equals zero with probability $e^{-\nu_2 y}(1-u)+(1-e^{-\nu_2 y})(1-w)$. Thus,

$$\begin{aligned}
&\mathbb{E}[e^{-s_2 W_1(k+1)}|R_1(k)=y] \\
&= \left(e^{-\nu_2 y}u + (1-e^{-\nu_2 y})w\right)\frac{\nu_2}{\nu_2+s_2} + e^{-\nu_2 y}(1-u) + (1-e^{-\nu_2 y})(1-w) \\
&= e^{-\nu_2 y}\frac{\sigma_2\nu_2 + (\sigma_1+\nu_2)(\nu_2+s_2)}{(\sigma_1+\sigma_2+\nu_2)(\nu_2+s_2)} + (1-e^{-\nu_2 y})\frac{(\sigma_1+\sigma_2)\nu_2 + (\nu_2+s_2)\nu_2}{(\sigma_1+\sigma_2+\nu_2)(\nu_2+s_2)}.
\end{aligned}$$

One can now compute $\mathbb{E}[e^{-(s_1 D_1(k)+s_2 D_1(k+1))}]$ using (5.32). By differentiation, we obtain the moments of D_1 and the autocovariance

$$\mathrm{Cov}[D_1(k), D_1(k+1)] = \frac{\sigma_1\sigma_2}{(\sigma_2+\nu_1)^2\nu_2(\sigma_1+\sigma_2+\nu_2)}. \tag{5.33}$$

The correlation coefficient between $D_1(k)$ and $D_1(k+1)$ is now obtained by dividing this expression by the variance of the stationary downtime D. Now that the first two moments of the stationary downtime distribution, as well as the correlation coefficient, are known, we can approximate the queue length of Q_1 in the extended machine repair model with the result on the queue length in the single-server model.

REMARK 5.4.1. The covariance as given in (5.33) and the resulting correlation coefficient both evaluate to zero when σ_1 or σ_2 is zero, or when σ_2, ν_1 or ν_2 tends to infinity. If either σ_1 or σ_2 is zero, one of the machines essentially never breaks down and there is no interference between the machines. When σ_2 tends to infinity, there is no correlation in the downtimes of M_1 either, since M_2 is practically always down. Therefore, every single downtime of M_1 will consist of a repair time of M_1 plus a residual repair time of M_2, which are both independent of anything else. When ν_1 tends to infinity, M_1 essentially does not require any repair time from the repairman and M_2 will never have to wait for the repairman to become idle. As a result, the downtimes of M_2 are independent. A waiting time for M_1 then comes down to either zero when M_2 is up, or the residual part of an M_2 repair. As the points in time at which a repair of M_2 is initiated are not biased by the breakdowns of M_1, the downtimes of M_1 are independent as well in that case. Equivalently, when ν_2 tends to infinity, M_2 does not require any repair time from the repairman, which means that downtimes of M_2 do not influence downtimes of M_1. As a result, there is no correlation in the downtimes of M_1 in this case either.

REMARK 5.4.2. For the case $\sigma_1 = \sigma_2$ and $\nu_1 = \nu_2$, an expression for the Laplace-Stieltjes transform $\mathbb{E}[e^{-s_1 W_1(k)}]$ can also be obtained using the arrival theorem (cf. [156]), which states that in a closed queueing network, the stationary state probabilities at instants at which customers arrive at a service unit are equal to the stationary state probabilities at arbitrary times for the network with one less customer. This implies that the probability distribution of the state of M_2 (either up or in repair) at a time M_1 breaks down is equal to the steady-state distribution of the state of M_2 in a system with $\sigma_1 = 0$, but with σ_2 and ν_2 left unchanged. In such a system, M_2 is the only machine requiring attention of the repairman, which greatly simplifies the analysis.

5.4.2 Choosing the appropriate dependence functions

In order to use Approximation 5.3.3 as an approximation for the probability generating function corresponding to L_i in the extended machine repair model, we need to identify suitable expressions for the functions $\chi_i(s)$ and $g_i(s)$. These functions need to match the dependence in the downtimes of M_i as well as possible; i.e. the expressions for (5.5) and (5.32) need to agree as much as possible. The quality of the choices for the functions directly influences the accuracy of the approximation, as they are the only source of error introduced. In order to obtain suitable expressions for $\chi_i(s)$ and $g_i(s)$, we perform two-moment fits commonly used in literature. To this end, the first two moments of the distributions represented by the Laplace-Stieltjes transforms $\chi_i(s)$ and $e^{-g_i(s)}$ must be determined. We do this based on expressions for $\chi_i'(0)$, $\chi_i''(0)$, $g_i'(0)$ and $g_i''(0)$, which we obtain by combining (5.3) and (5.6) with results for the first two moments of the downtime distribution and the correlation coefficient of the consecutive downtimes. These depend on the distributions of the repair times R_1 and R_2, among others. For exponential repair-time distributions, the results required were obtained in Section 5.4.1 by inspection of the embedded discrete-time Markov chain $\{\boldsymbol{X}_n, n \geq 0\}$. By using the same methods, similar results can be obtained for phase-type repair times.

5.4.2.1 Obtaining derivatives of the dependence functions

To obtain values for $\chi_i'(0)$, $\chi_i''(0)$, $g_i'(0)$ and $g_i''(0)$, we solve a set of equations. In Section 5.4.1, we have expressed $\mathbb{E}[D_i]$, $\mathbb{E}[D_i^2]$ and $\lim_{k\to\infty} \mathbb{E}[D_i(k)D_i(k+1)]$ in terms of the parameters of the extended machine repair model. By (5.3) and (5.6), we have that these expressions are related to the functions $\chi_i(\cdot)$ and $g_i(\cdot)$ as follows:

$$\begin{aligned}
\mathbb{E}[D_i] &= \frac{\chi_i'(0)}{g_i'(0)-1}, \\
\mathbb{E}[D_i^2] &= \frac{\chi_i''(0) - \mathbb{E}[D_i](2\chi_i'(0)g_i'(0) + g_i''(0))}{1 - g_i'(0)^2}, \\
\mathbb{E}[D_i(k)D_i(k+1)] &= -\chi_i'(0)\mathbb{E}[D_i] + g_i'(0)\mathbb{E}[D_i^2]. \qquad (5.34)
\end{aligned}$$

These three equations in four unknowns fix values for $\chi_i'(0)$ and $g_i'(0)$, but leave one degree of freedom in the determination of $\chi_i''(0)$ and $g_i''(0)$. This freedom can be used to fine-tune the model. For example, one might assume the independent component of the downtime to be distributed according to a certain distribution. This would lead to an additional equation for $\chi_i''(0)$ in terms of $\chi_i'(0)$, which then also fixes values for $\chi_i''(0)$ and $g_i''(0)$.

5.4.2.2 Expressions for the dependence functions

We now determine suitable expressions for $\chi_i(\cdot)$ and $g_i(\cdot)$. For this purpose, there are many approaches possible. Below, we base the choices of $\chi_i(\cdot)$ and $g_i(\cdot)$ on two-moment approximations. To apply these two-moment approximations, we use the squared coefficient of variation, which for a random variable Z is defined as $c_Z^2 = \mathrm{Var}[Z]/\mathbb{E}[Z]^2 = \frac{\mathbb{E}[Z^2]}{\mathbb{E}[Z]^2} - 1$.

Using the derivatives of the dependence function, we obtain the first two moments of the distributions represented by the Laplace-Stieltjes transforms $\chi_i(s)$ and $e^{-g_i(s)}$. As explained in Section 5.2, the function $\chi_i(s)$ is the Laplace-Stieltjes transform corresponding to a random variable representing the independent component of the downtime. The first two moments of this component are given by $-\chi_i'(0)$ and $\chi_i''(0)$, respectively, and the squared coefficient of variation is consequently given by $\frac{\chi_i''(0)}{(\chi_i'(0))^2} - 1$. The function $e^{-g_i(s)}$ is the Laplace-Stieltjes transform of an infinitely divisible distribution, namely the distribution of the incremental component of $D(k+1)$ per unit of $D(k)$. The corresponding first two moments are given by $g_i'(0)$ and $(g_i'(0))^2 - g_i''(0)$, respectively, and therefore the squared coefficient of variation is given by $-\frac{g_i''(0)}{(g_i'(0))^2}$.

Based on the two moments and the squared coefficient of variation for each of the distributions, we employ commonly used distributional two-moment fit approximations as described in [238, pp. 358–360]. For instance, in case of a squared coefficient of variation smaller than one, one fits a mixture of an Erlang(k,γ) and an Erlang$(k-1,\gamma)$ distribution to the moments ($k \geq 2, \gamma > 0$), whereas for a squared coefficient of variation larger than one, one uses a hyperexponential distribution with two phases and balanced means. In the special case of a squared coefficient of variation of zero or one, one uses a deterministic or exponential distribution, respectively. The parameters for each of these distributions are based on the first two moments, which are given as an input for this procedure.

Thus, we choose the functions $\chi_i(s)$ and $g_i(s)$ as follows. First, we compute the moments (cf. Section 5.4.1), which we use in (5.34) to find the first two derivatives of $\chi_i(s)$ and $g_i(s)$. Based on these derivatives, we then fit repair-time distributions using the two-moment approximations in [238, pp. 358–360]. Recall that we assumed in Section 5.2 that $g_i(s)$ has a completely monotone derivative, so that the Laplace-Stieltjes transform $e^{-g_i(s)}$ represents an infinitely divisible distribution. The distributions mentioned above satisfy this assumption:

- For a deterministic distribution with value x and Laplace-Stieltjes transform e^{-sx}, we have $g_i(s) = sx$. This function obviously has a completely monotone derivative, since $\frac{d}{ds} g_i(s) = x \geq 0$ and $\frac{d^n}{ds^n} g_i(s) = 0$ for all $n \geq 2$.
- For an exponential distribution and a H_2 distribution, see [96, p. 452] on mixtures of exponential distributions.
- A mixture of an Erlang(k,γ) distribution and an Erlang$(k-1,\gamma)$ distribution with weights $q \in [0,1]$ and $1-q$, respectively, results in the Laplace-Stieltjes transform $q\left(\frac{\gamma}{\gamma+s}\right)^k + (1-q)\left(\frac{\gamma}{\gamma+s}\right)^{k-1}$. Hence, the function $g_i(s) = -\log\left(q\left(\frac{\gamma}{\gamma+s}\right)^k + (1-q)\left(\frac{\gamma}{\gamma+s}\right)^{k-1}\right)$. Furthermore, we have that

$$\frac{d^n}{ds^n} g_i(s) = (-1)^{n+1}(n-1)!\left(\frac{k}{(\gamma+s)^n} - \frac{(1-q)^n}{(\gamma+(1-q)s)^n}\right).$$

The second term $(n-1)!$ is positive. The third term is also positive, since we have that $(\gamma+s)^n \frac{(1-q)^n}{(\gamma+(1-q)s)^n} \leq \frac{(\gamma+(1-q)s)^n}{(\gamma+(1-q)s)^n} = 1 < 2 \leq k$. Therefore, derivatives of odd order are positive through the first term. Similarly, we have that derivatives of even order are negative. Hence, $g_i(s)$ has a completely monotone derivative.

5.4.3 Resulting approximation

Now that we have obtained suitable expressions for $\chi_i(s)$ and $g_i(s)$, Approximation 5.3.3 also directly yields an approximation for the (probability generating function of the) marginal queue length distribution in the extended machine repair model.

APPROXIMATION 5.4.1. *In the extended machine repair model, an approximation $L_{i,app}$ for the queue length of Q_i is given by the probability generating function*

$$\mathbb{E}[z^{L_{i,app}}] = p_{up}\mathbb{E}[z^L|\textit{server up}] + p_{down}\mathbb{E}[z^L|\textit{server down}], \tag{5.35}$$

where expressions for $\mathbb{E}[z^L|\textit{server up}]$, $\mathbb{E}[z^L|\textit{server down}]$, p_{up} and p_{down} are given by (5.19)*,* (5.27) *and* (5.29)*, respectively, but with λ, $\widetilde{B}(\cdot)$, σ, $\chi(\cdot)$ and $g(\cdot)$ replaced by the extended machine repair model counterparts λ_i, $\widetilde{B}_i(\cdot)$, σ_i, $\chi_i(\cdot)$ and $g_i(\cdot)$.*

REMARK 5.4.3. Note that (5.32) cannot be rewritten in the form of the two-dimensional Laplace-Stieltjes transform (5.5); i.e. the dependence structure we assumed in (5.1) or (5.5) does not perfectly model the distribution and the interdependence of the downtimes of M_i. In addition to this modelling approximation and the approximation error discussed in Section 5.3.1.6, a numerical approximation error is introduced by truncation of the infinite product in (5.4). However, the latter error can be made negligibly small.

5.4.4 Approximations for generalisations of the model

In the previous sections, we derived an approximation for the extended machine repair model with two machines and a single repairman. However, the approach followed can be readily extended to approximate queue lengths of first-layer queues in an equivalent model with a larger number of queues and machines or multiple repairmen. Moreover, the approach followed in Section 5.4.1 for deriving the moments and the correlation coefficient of the downtimes remains valid when assuming phase-type repair time distributions. We discuss these model generalisations below. Note that in the cases below, we apply the analysis to the single-server model as given in Section 5.3 without any modification.

Larger numbers of machines and first-layer queues When we generalise the extended machine repair model as described in Section 5.2 to allow for $N > 2$ machines $M_1, \ldots, M_N$ and thus N first-layer queues $Q_1, \ldots, Q_N$, we can still use Approximation 5.4.1 like before to approximate the probability generating functions of $L_1, \ldots, L_N$. The approach for deriving appropriate functions for $\chi_i(s)$ and $g_i(s)$, $i = 1, \ldots, N$, needed to use Approximation 5.4.1, remains largely the same. However, by introducing a larger number of machines, the computation of the first two moments and the correlation coefficient of downtimes in the extended machine repair model becomes increasingly cumbersome. As opposed to the case $N = 2$ as assumed in Section 5.4.1, the repair buffer can now contain multiple machines. Since the repair facility serves the queue in a first-come-first-served

manner, the order in which the machines are waiting for repair needs to be included in the state space of the embedded discrete-time Markov chain describing the states of the machines. Subsequently, considerably more conditioning is needed to compute the terms $\mathbb{E}[e^{-s_1 W_1(k)}]$ and $\mathbb{E}[e^{-s_2 W_1(k+1)}|R_1(k) = y]$ in (5.32) and, ultimately, the moments and the correlation coefficient of the downtimes.

Multiple repairmen In the extended machine repair model, it is assumed there is only one repairman assigned to repair machines. This assumption can be relaxed to allow for $K > 1$ repairmen in the repair facility, each working on a different machine and taking the broken machines out of the repair buffer in a first-come-first-served manner. When $K \geq N$, a broken machine will always be taken into repair immediately. As a result, machines do not compete for repair facilities anymore, and consecutive downtimes of a machine become independent. Therefore, when taking $\chi_i(s)$ such that it equals the Laplace-Stieltjes transform of the repair-time distribution of M_i and taking $g_i(0) = 0$, the *exact* probability generating function of the distribution of L_i is given by Approximation 5.4.1. When $N > K$, consecutive downtimes of the machine remain correlated. Again, the approximation as developed in this chapter remains valid, but difficulties arise in deriving the appropriate functions for $\chi_i(s)$ and $g_i(s)$, $i = 1, \ldots, N$. More specifically, the computation of the moments and the correlation coefficient of the consecutive downtimes of each of the machines again becomes increasingly complicated. Since machines can now be repaired simultaneously, the order in which machines return to service after repair is not necessarily the same as the order in which machines break down. This introduces extra conditioning in the computation of $\mathbb{E}[e^{-s_2 W_1(k+1)}|R_1(k) = y]$ in (5.32), since the machines which were already waiting for repair at the start of $W_1(k)$ may not have returned to an operational state by the time $R_1(k)$ has passed. This evidently influences $W_1(k+1)$.

Phase-type distributed repair times In Section 5.4.1, we derived an explicit expression for the correlation coefficient of consecutive downtimes of a machine, in case repair times are exponentially distributed. For phase-type repair-time distributions, a similar approach for studying the embedded discrete-time Markov chain can be followed to obtain the numbers needed to construct the functions $\chi_i(s)$ and $g_i(s)$ in Section 5.4.2. The computations may become more involved, but remain conceptually the same. This leads to a more complicated expression for $\mathbb{E}[e^{-s_1 W_1(k)}]$ in (5.32). For the computation of $\mathbb{E}[e^{-s_2 W_1(k+1)}|R_1(k) = y]$, extra conditioning on the repair phase is also needed.

5.5 Numerical study

We now give some numerical examples to assess the accuracy of Approximation 5.4.1. In Section 5.5.1, we compare our approximation for the marginal queue length to simulation results for a typical setting. Then, in Section 5.5.2, we observe the effect of the model parameters and identify several key factors determining the accuracy of the approximation.

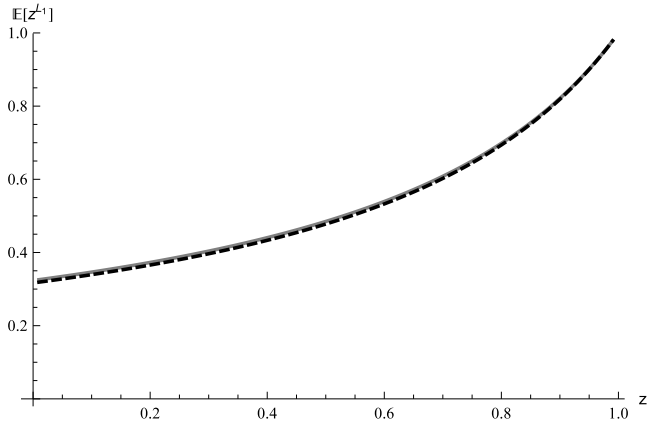

FIGURE 5.3: Plot of $\mathbb{E}[z^{L_{1,app}}]$ (solid curve) and $\mathbb{E}[z^{L_1}]$ (dashed curve).

5.5.1 Initial glance at the approximation

Consider a system where $\lambda_1 = 0.25$, $\sigma_1 = \sigma_2 = 1$ and B_1, R_1 and R_2 are exponentially (1) distributed. Note that the settings for λ_2 and B_2 do not influence the length of Q_1. In Figure 5.3, we plot the probability generating function corresponding to $L_{1,app}$, which is given in (5.35), and the probability generating function corresponding to L_1, which we obtained by simulation.

We observe in this figure that $\mathbb{E}[z^{L_{1,app}}]$ matches $\mathbb{E}[z^{L_1}]$ very closely. The error made is largest at $z = 0$, where $\mathbb{E}[z^{L_{1,app}}]$ is 2.09% larger than the value of $\mathbb{E}[z^{L_1}]$. As for the expectation of the queue length, we have that $\mathbb{E}[L_{1,app}] = \frac{d}{dz}\mathbb{E}[z^{L_{1,app}}]|_{z=1} = 2.205$, while the theoretical mean $\mathbb{E}[L]$ equals 2.220. This is a typical performance of the approximation. As we will see in the next section, the accuracy of the approximation can become worse if the downtimes in the extended machine repair model are extraordinarily correlated. Nonetheless, in realistic systems, even in the worst-case scenarios, the difference in the expected queue lengths is not much more than 10%.

5.5.2 Accuracy of the approximation

We now turn to the study of the parameter effects on the accuracy of the approximation. As we will see, the approximation performs very well over a wide range of parameter settings. We also observe several parameter effects.

To study the accuracy of the approximation, we compare the approximated values for the mean of L_1 with the values obtained by numerical methods such as simulation or the power-series algorithm (cf. Chapter 2) in various instances of the extended machine repair model. We regard instances where B_1 is exponentially (μ_1) distributed, and R_1 and R_2 are exponentially distributed with rates ν_1 and ν_2, respectively. In fact, we use the same test bed as the one we used to assess the accuracy of Approximation 4.2.1 in Section 4.2.2. Thus, the instances we use to test the accuracy of the distributional approximation are given by the combinations of the parameter values listed in Table 4.1.

For each of these systems, we compare the approximated mean queue lengths of the

TABLE 5.1: Percentual relative errors Δ of the mean queue length approximation categorised in bins.

	0-0.01%	0.01-0.1%	0.1-1%	1-5%	>5%+
% of rel. errors Δ	25.93%	32.30%	32.15%	9.63%	0.00%

first queue, namely $\mathbb{E}[L_{1,app}] = \frac{d}{dz}\mathbb{E}[z^{L_{1,app}}]|_{z=1}$, to the actual mean queue length $\mathbb{E}[L_1]$. Subsequently, we again compute the relative error of these approximations, i.e.

$$\Delta = 100\% \times \left| \frac{\mathbb{E}[L_{1,app}] - \mathbb{E}[L_1]}{\mathbb{E}[L_1]} \right|. \tag{5.36}$$

In Table 5.1, the resulting relative errors are summarised. We note that none of these errors is greater than 5% and that the majority of these errors does not exceed 0.1%. This seems to remain the case even as the load goes to one or for extreme values of the imbalance in the system. These results show that Approximation 5.4.1 works very well for typical systems. Comparing Table 5.1 with the results from Section 4.2.2, the approximation does not challenge the accuracy of the light-traffic approximation (and as a consequence neither that of the interpolation approximation) derived in Chapter 4. The added value of the distributional approximation, however, lies in the fact that it approximates the entire distribution rather than merely its first moment and thus more performance measures can be evaluated.

To observe any parameter effects, we also give the mean relative error categorised in some of the variables in Table 5.2. From Table 5.2(a), we conclude that the accuracy of the approximation is not very sensitive to the load of the queue. Based on Tables 5.2(b) and 5.2(c), however, we note that the orders of magnitude of the breakdown and repair rates do impact the accuracy of the approximation. This is due to the fact that the rate at which products move (i.e. arrive and get served) with respect to the life and repair times of the machine differ in these cases. In Tables 5.2(d) and 5.2(e), we see that the imbalance of the breakdown and repair rates do impact the accuracy as well (but to a lesser extent). We conclude this chapter by discussing the observed effects in more detail below.

Effect of fast moving products In Table 5.2, we observe that decreasing the uptimes and repair times of the machines relative to the movement speed of the products leads to a decrease in the performance of the approximation. In other words, when the movement speed of products (i.e. arrival rate and service rate) increases with respect to the breakdown rates and repair rates of the machines, the performance of the approximation deteriorates. To further examine this effect, we regard the queue length of Q_1 in systems with arrival rates ranging from $\lambda_1 = 0$ to $\lambda_1 = 3$ and an exponentially distributed service time B_1 with rate $10\lambda_1/3$ varying accordingly so as to keep the load fixed. Furthermore, the breakdown rates are given by $\sigma_1 = \sigma_2 = 1$ and the repair times R_1 and R_2 are exponentially (1) distributed. After applying Approximation 5.4.1 to the mean queue length of Q_1 in these systems and comparing it with exact results, we obtain Figure 5.4, where the relative error Δ (see (5.36)) is given as a function of λ_1. We indeed observe that the faster the products arrive (and get served), the more inaccurate the approximation becomes. This effect can be explained by the fact that faster moving products are

TABLE 5.2: Mean percentual relative error Δ categorised in ρ_1 (a), the variables controlling the order of magnitude of σ_i and ν_i, namely a_i^σ (b) and a_i^ν (c), and the variables controlling the imbalance, b_j^σ (d) and b_j^ν (e).

(a)

ρ_1	0.25	0.5	0.75
Mean rel. error Δ	0.328%	0.316%	0.335%

(b)

a_i^σ	0.1	1	10
Mean rel. error Δ	0.564%	0.294%	0.121%

(c)

a_i^ν	0.1	1	10
Mean rel. error Δ	0.727%	0.219%	0.033%

(d)

b_j^σ	(1, 1)	(1, 2)	(2,1)	(1, 5)	(5, 1)
Mean rel. error Δ	0.354%	0.275%	0.414%	0.149%	0.439%

(e)

b_j^ν	(1, 1)	(1, 2)	(2,1)	(1, 5)	(5, 1)
Mean rel. error Δ	0.395%	0.344%	0.143%	0.212%	0.537%

more sensitive to variations caused by dependence in the downtimes. A small increase in the downtime causes more additional products to build up in the queue, while such an increase may even remain unnoticed in case of slow products with long interarrival times. Hence, in the former case, the error made in approximating the dependence structure of consecutive downtimes by the functions $\chi_1(\cdot)$ and $g_1(\cdot)$ shows itself more in the approximation of the mean queue length than in the latter case.

Effect of the degree of dependence From Table 5.2, it is apparent that the accuracy of the approximation is influenced by the values for b_j^σ and b_j^ν. This can be mainly explained by the fact that these values determine the strength of the dependence between consecutive downtimes in M_1. To illustrate this effect, let us observe systems where B_1, as well as both R_1 and R_2, is exponentially (1) distributed. Moreover, we have $\lambda_1 = 1/4$ and $\sigma_1 = 1$. In Figure 5.5, we show the relative error Δ in approximating the mean queue length of Q_1 as a function of σ_2. Since the breakdown rate of M_2 varies in these systems, the strength of the dependence changes accordingly. In Figure 5.5, r_{scaled}, the correlation coefficient of consecutive downtimes as computed in Section 5.4.1, is given in a scaled form so as to fit the graph. We see that the accuracy of the approximation is, at least in this case,

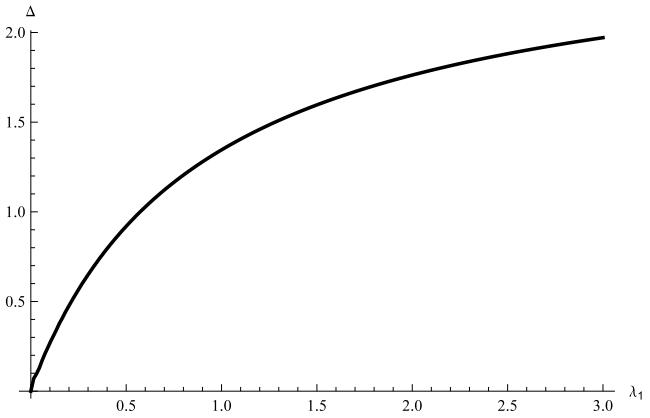

FIGURE 5.4: The percentual relative error Δ made as a function of the products' arrival rate.

largely determined by the strength of the correlation between the downtimes. Intuitively this makes sense. In case there is no such correlation in the model (for example, when $\sigma_2 = 0$ or $\sigma_2 \uparrow \infty$), the approximation should at least be close to being exact. Using the procedure of Section 5.4.2, $g_1(\cdot)$ will resolve to zero in such a case. When $\chi_1(\cdot)$ is chosen to match $\widetilde{D}(\cdot)$, the approximation becomes exact, as the assumed downtime structure in (5.1) with the functions $\chi_1(\cdot)$ and $g_1(\cdot)$ will then describe the dependence in an exact way.

Effect of the variability of the repair times Table 4.1 only includes instances of the extended machine repair model for which repair times are exponentially distributed. In practice, however, the level of variability in the repair times may be much higher. To investigate whether the accuracy of Approximation 5.4.1 is influenced by this, we again study the instance of the model as presented in Section 5.5.1. However, we now assume the repair times R_1 and R_2 to be hyperexponentially distributed with mean one. In particular, we study the behaviour of the relative error made by the approximation as the squared coefficients of variation of R_1 and R_2 ($c^2_{R_1}$ and $c^2_{R_2}$) increase. Figure 5.6 shows the relative error (as defined in (5.36), however now with the sign included) in approximating $\mathbb{E}[L_1]$ versus the squared coefficient of variation of the repair times (which we assume to be equal). The various parameter combinations for the hyperexponential repair-time distribution needed to match the squared coefficients of variation are chosen as described in [238, pp. 358–360]. The figure shows that even up to a squared coefficient of variation of 8, which represents highly variable repair times for both machines, the error made is only approximately 1%. Therefore, the accuracy of Approximation 5.4.1 seems to remain very high even for repair times with very high variability.

Comparison with Wartenhorst's approximation in [269] The approach that we used in this chapter to approximate the queue length distributions of the first-layer queues involves the study of the dependence between consecutive downtimes in the second layer of the model. As mentioned before, the extended machine repair model has also been

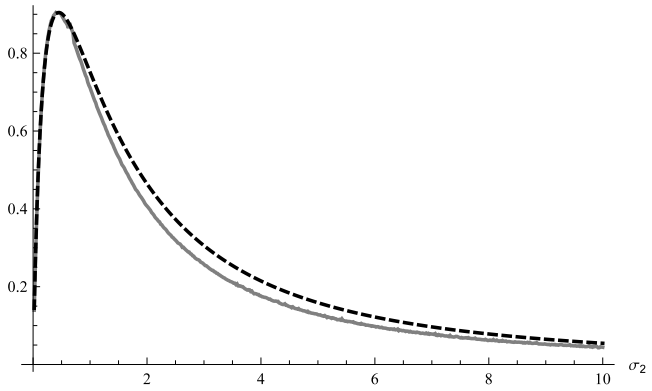

FIGURE 5.5: The percentual relative error made, Δ (solid curve), and the scaled value of the correlation coefficient, r_{scaled} (dashed curve), as a function of the breakdown rate of M_2.

studied by Wartenhorst in [269]. However, in [269], it is assumed that $\sigma_1 = \sigma_2$ and that R_1 and R_2 are exponentially distributed with equal rates. This results in $\widetilde{D}_1(\cdot) = \widetilde{D}_2(\cdot)$. In his study, Wartenhorst approximates the mean length of Q_1 with the mean queue length in a single-server vacation queue, where the distribution of the vacation lengths equals the stationary downtime distribution of M_i, but where the downtimes are assumed to be *completely independent*. The queue length distribution of this single-server queue is obtained by applying the Fuhrmann-Cooper decomposition (cf. [102]). Wartenhorst's approximation is exact by construction for a system where downtimes are independent, and accurate whenever downtimes are only slightly dependent. Although [269] assumes equal breakdown rates and identically distributed repair times for the machines, his approach can be extended with some effort to allow for cases where these assumptions are violated.

To compare the accuracy of the approximation derived in the present chapter with that of [269], we study a set of systems with highly dependent downtimes. For these systems, we assume that $\sigma_1 = 100$, $\sigma_2 = 0.02$ and that R_2 is exponentially distributed with rate 0.01. To maximise the correlation in the downtimes of M_1, we assume R_1 to be hyperexponentially distributed with probability parameters 0.975 and 0.025 and rate parameters 100 and 0.01. The value for the correlation coefficient in these systems evaluates to 0.26. We vary λ_1 between 0 and 0.01. Furthermore, we assume B_1 to be exponentially distributed with rate $500\lambda_1$ so as to keep the load at Q_1 fixed.

In Figure 5.7, the relative error Δ in approximating $\mathbb{E}[L_1]$ is given for both the approximation obtained in this chapter and Wartenhorst's approximation. We see the same effect of fast moving products as before. The faster the products move, the less accurate both approximations become. However, we see that the degree of dependence has a significantly larger effect on the accuracy of Wartenhorst's approximation than on that of the approximation presented here. Since the degree of the dependence between the downtimes is the major source of inaccuracy for both approximations (cf. Section 5.3.4), one could conclude that Approximation 5.4.1 performs as well as Wartenhorst's approximation in cases with only slight dependences and better in cases with stronger correlations

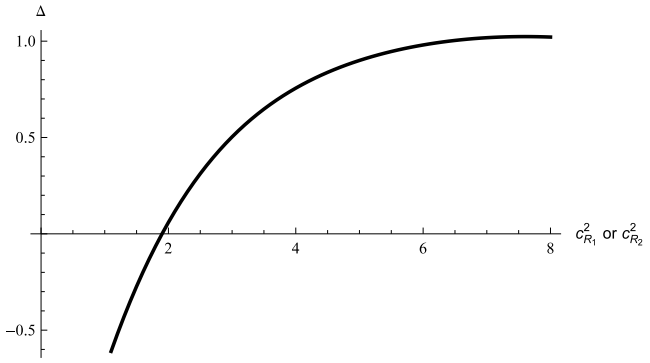

FIGURE 5.6: The accuracy of the approximation as a function of the squared coefficient of variation of the repair times R_1 and R_2.

between the downtimes. This observation shows that the dependence between the layers cannot be ignored.

Appendix

5.A Proof of Lemma 5.3.1

PROOF. The function $E(z)$ is continuous on $[0, \widetilde{\mu}(\sigma))$. We also have that $1-E(0) = 1$ and $\lim_{z\to\widetilde{\mu}(\sigma)} 1-E(z) = -\infty$. Hence, there exists at least one root in $(0, \widetilde{\mu}(\sigma))$ by Bolzano's theorem.

To prove that there is at most one root in $(0, \widetilde{\mu}(\sigma))$, we show that $1-E(z)$ is strictly decreasing in z. In other words, we show that $E(z)$ is strictly increasing in z by studying the monotonicity of each of the terms in (5.15) separately. First, since $\chi(\cdot)$ is the Laplace-Stieltjes transform of (the distribution of) a positive and continuous random variable (see Section 5.2), it is a strictly decreasing function. Recalling that $\lambda > 0$, this means that the first term $\chi(\lambda(1-z))$ is therefore strictly increasing in z. For the monotonicity of the second term $A^2(z)$, we show that $A(z)$ is strictly decreasing (i.e. $A'(z) < 0$ for all values of z considered). We have that

$$\begin{aligned} A'(z) = {} & \frac{\sigma\lambda}{(\sigma+\lambda(1-z))^2}\frac{z(1-\widetilde{B}(\sigma+\lambda(1-z)))}{z-\widetilde{B}(\sigma+\lambda(1-z))} \\ & + \frac{\sigma}{\sigma+\lambda(1-z)}\Bigg(\frac{(1-\widetilde{B}(\sigma+\lambda(1-z)))+z\lambda\widetilde{B}'(\sigma+\lambda(1-z))}{z-\widetilde{B}(\sigma+\lambda(1-z))} \\ & \qquad - \frac{z(1-\widetilde{B}(\sigma+\lambda(1-z)))(1+\lambda\widetilde{B}'(\sigma+\lambda(1-z)))}{(z-\widetilde{B}(\sigma+\lambda(1-z)))^2}\Bigg). \end{aligned} \tag{5.37}$$

Since $\widetilde{B}(\cdot)$ is a Laplace-Stieltjes transform representing a positive, continuous random variable, we have that $1-\widetilde{B}(\sigma+\lambda(1-z)) > 0$ and $\widetilde{B}'(\sigma+\lambda(1-z)) > 0$, which also readily implies that $z\lambda\widetilde{B}'(\sigma+\lambda(1-z)) > 0$ and $1+\lambda\widetilde{B}'(\sigma+\lambda(1-z)) > 0$. This means

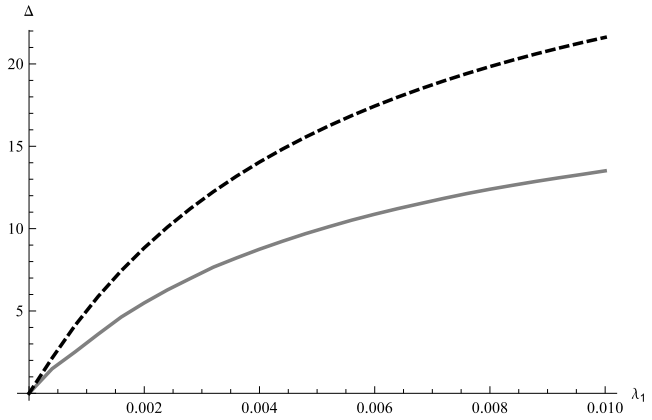

FIGURE 5.7: The percentual relative error made by Approximation 5.4.1 (solid curve) and Wartenhorst's approximation (dashed curve).

that in (5.37), the numerator of the second fraction in the first term and the numerators of the fractions between the brackets are all positive. Moreover, we have that $z - \widetilde{B}(\sigma + \lambda(1-z)) < 0$ for all $z \in (0, \widetilde{\mu}(\sigma))$, which consequently implies through the denominators that the second fraction of the first term and the expression between the brackets each are negative. Combining this with the fact that evidently both $\sigma/(\sigma + \lambda(1-z))$ and $\sigma\lambda/(\sigma + \lambda(1-z))^2$ are positive as $z < 1$, we have that $A'(z) < 0$ and thus that the second term $A^2(z)$ is strictly increasing. For the third term $\widetilde{D}(\lambda(1-z) + g(\lambda(1-z)))$, we have that $\lambda(1-z) + g(\lambda(1-z))$ is strictly decreasing in z, as $g(s)$ is increasing in s. The latter is the case, since $e^{-g(s)}$ is the Laplace-Stieltjes transform representing a positive, continuous random variable and therefore strictly decreasing in s. Furthermore, the Laplace-Stieltjes transform $\widetilde{D}(\cdot)$ is a strictly decreasing function. Therefore, the third term is strictly increasing in z.

Summarising, all of the terms of $E(z)$ as expressed in (5.15) are strictly increasing for the values of z considered. As a result, $E(z)$ itself is strictly increasing for $z \in (0, \widetilde{\mu}(\sigma))$. Therefore, the denominator $1-E(z)$ has exactly one root on the real line in $(0, \widetilde{\mu}(\sigma))$. □

6

OPTIMISATION OF QUEUE LENGTHS

The analysis of the extended machine repair model in the previous chapters reveals that the competition of the machines for the repairman's resources has a big impact on the first-layer queues. As the repairman repairs the machines on a first-come-first-served basis and only one at a time, there exist correlations in the machines' downtimes that may have a significant effect on the queue lengths in the first layer. This raises the question whether the repairman's strategies could be adapted in order to reduce these queue lengths. Motivated by this, we now drop the assumption that the repairman repairs the machine in the order of breakdown. Instead, we concern ourselves in this chapter with the dynamic control problem of how the repairman should allocate his resources to the machines at any point in time so that the long-term average (weighted) sum of the queue lengths of the first-layer queues is minimised. Since the optimal policy for the repairman cannot be found analytically, we propose a near-optimal policy. We do this by combining intuition and results from queueing theory with techniques from Markov decision theory. We study the relative value functions for several policies for which the model can be decomposed in less complicated subsystems, and we combine the results with the classical one-step policy improvement algorithm. The resulting policy is easy to apply, is scalable in the number of machines and performs very well for a wide range of parameter settings.

6.1 Introduction

We are concerned with the question of how the repairman should allocate his capacity dynamically to the machines at any point in time, given complete information on the lengths of the queues of products and the states of the machines at all times. We aim to formulate a policy that minimises the long-term average (weighted) sum of the queue lengths of the first-layer queues. To this end, we use several techniques from Markov decision theory. When formulating this problem as a Markov decision problem, one may be able to obtain the optimal policy numerically for a specific set of parameter settings by truncating the state space. However, due to the multi-dimensionality of the model, the computation time needed to obtain reliable and accurate results may be infeasibly long. Moreover, these numerical methods are cumbersome to implement, do not scale well in the number of dimensions of the problem, lack transparency and provide little insight into the effects of the model parameters. To overcome these problems, we derive

a near-optimal policy that can be expressed explicitly in terms of the model parameters by use of the one-step policy improvement method.

The one-step policy improvement method requires a relative value function of an initial policy which can be obtained analytically by solving the Poisson equations known from standard theory on Markov decision processes. The result is then used in a single step of the policy iteration algorithm from Markov decision theory to obtain an improved policy. Although the relative value function of the improved policy is usually hard to compute, the improved policy itself is known explicitly as a function of the state and the parameters of the model. Moreover, it is known that the improved policy performs better in terms of costs than the initial policy, so that it may be used as an approximation for the optimal policy. The intrinsic idea of one-step policy improvement goes back to Norman [181]. Since then, this method has been successfully applied to derive nearly optimal state-dependent policies in a variety of applications, such as the control of traffic lights [113], production planning [276] and the routing of telephone calls in a network or call center [36, 185, 218].

In this chapter, we apply the one-step policy improvement method in a more structured way. It is tempting to construct an improved policy based on a policy that creates the simplest analytically tractable relative value function. However, this improved policy might not have the best performance or may result in unstable queues. In our case, we start with multiple policies that are complementary in different parameter regions in both of these aspects and combine them to come up with an improved policy that is more broadly applicable. In our objective to derive a near-optimal policy over a broad range of parameter values, we need to start with initial policies that do not defy the derivation of closed-form expressions for their corresponding relative value functions. In some cases, we use insights from queueing theory to provide an accurate approximation for the relative value function and the long-term averaged costs. We use these results to construct a near-optimal policy that requires no computation time, is easy to implement and gives insights into the effects of the model parameters.

Section 6.2 gives a mathematical description of the control problem and introduces the notation required. Although the optimal policy for this control problem cannot be obtained explicitly, several of its structural properties can be derived. As we will see in Section 6.3, the optimal policy makes the repairman work at full capacity whenever there is at least one machine down and behaves like a threshold policy. Subsequently, we focus on finding a policy which generally performs nearly as well as the optimal policy. As input for the one-step policy improvement algorithm, we study two policies in Section 6.4 for which the system decomposes into multiple subsystems, so that the system becomes easier to evaluate. The first of these policies, which we will call the static policy, always reserves a certain predetermined fraction of repair capacity to each machine regardless of the state of the machines. Therefore, the machines behave independently of each other under this policy, which allows us to derive an exact expression for the relative value function. As the static policy cannot always be used as an input for the one-step policy improvement algorithm due to instability issues, we also study a second class of policies in Section 6.4. More specifically, we study the priority policy, in which the repairman always prioritises the repair of a specific machine over the other when both machines are down. Under this policy, the repairman assigns his full capacity to the high-priority machine when it is down irrespective of the state of the low-priority machine. This makes the system easier to analyse. Nevertheless, it is hard to obtain the relative value function for this policy

exactly, but we are able to identify most of its behaviour. Although analytic results on the relative value functions of these policies are of independent interest, we use these results in Section 6.5 in combination with the one-step policy improvement algorithm. This ultimately results in a well-performing and nearly optimal policy, which is given in terms of a few simple decision rules. The resulting policy turns out to be scalable in the number of machines and corresponding first-layer queues in the model, so that the policy can be readily extended to allow for a number of machines larger than two. Finally, extensive numerical results in Section 6.6 show that the proposed policy is highly accurate over a wide range of parameter settings. Based on these numerical results, we identify the key factors determining the performance of the near-optimal policy.

6.2 Problem formulation and notation

Again, we follow the majority of the model assumptions and notation as introduced in Sections 1.3.1 and 2.2. In particular, we assume that the uptime of machine M_i is exponentially (σ_i) distributed, after which it requires an exponentially (ν_i) amount of service from the repairman before it is able to continue processing products. The fundamental difference with the previous chapters is that we no longer assume the repairman to repair the machines in the order of breakdown. In fact, the machines share the capacity of the repairman. At any moment in time, the repairman is able to decide how to divide his total repair capacity over the machines. More specifically, he can choose the fractions of capacity q_1 and q_2 that are allocated to the repair of M_1 and M_2, respectively, so that the machines are being repaired at rate $q_1\nu_1$ and $q_2\nu_2$, respectively. We naturally have that $0 \leq q_1+q_2 \leq 1$ and that $q_i = 0$ whenever M_i is operational. The objective is to allocate the repair capacity dynamically in such a way that the average long-term weighted number of products in the system is minimised.

In order to describe this dynamic optimisation problem mathematically, one does not only need to keep track of the queues of products, but also of the conditions of the machines. To this end, we define the state space of the system as $\mathscr{S} = \mathbb{N}^2 \times \{0,1\}^2$. Each possible state corresponds to an element $\boldsymbol{s} = (x_1, x_2, w_1, w_2)$ in $\mathscr{S}$, where x_1 and x_2 denote the number of products in Q_1 and Q_2, respectively. The variables w_1 and w_2 denote whether M_1 and M_2 are in an operational (1) or in a failed state (0), respectively. Note that this state space is different from the one introduced in Section 2.2, as there is no need to keep track of the order in which the machines broke down due to the lack of a first-come-first-served assumption.

The repairman bases his decision on the information $\boldsymbol{s}$, and therefore any time the state changes can be regarded as a decision epoch. At these epochs, the repairman takes an action $\boldsymbol{a} = (q_1, q_2)$ out of the state-dependent action space $\mathscr{A}_{\boldsymbol{s}} = \{(q_1,q_2) : q_1 \in [0, 1-w_1] \wedge q_2 \in [0, 1-w_2] \wedge q_1 + q_2 \leq 1\}$, where q_i denotes the fraction of capacity assigned to M_i, $i = 1,2$. The terms $1-w_1$ and $1-w_2$ included in the description of the action set enforce the fact that the repairman can only repair a machine if it is down. Now that the states and actions are defined, we introduce the cost structure of the model. The objective is modelled by the cost function $c(\boldsymbol{s},\boldsymbol{a}) = c_1x_1 + c_2x_2$, where c_1 and c_2 are non-negative real-valued weights. Thus, when the system is in state $\boldsymbol{s}$, the weighted number of customers present in the system equals $c(\boldsymbol{s},\cdot)$ regardless of the action $\boldsymbol{a}$ taken by the repairman.

With this description, the control problem can be fully described as a Markov decision problem. To this end, we uniformise the system (see e.g. [162]); i.e. we add dummy transitions (from a state to itself) such that the outgoing rate of every state equals a constant parameter γ, the uniformisation parameter. We choose $\gamma = \lambda_1 + \lambda_2 + \mu_1 + \mu_2 + \sigma_1 + \sigma_2 + \nu_1 + \nu_2$ and we assume that $\gamma = 1$ without loss of generality, since we can always achieve this by scaling the model parameters. Note that this assumption has the benefit that rates can be considered to be transition probabilities, since the outgoing rates of each state sum up to one. Thus, for $i = 1, 2$, any action $\boldsymbol{a} \in \mathscr{A}_{\boldsymbol{s}}$ and any state $\boldsymbol{s} \in \mathscr{S}$, the transition probabilities p are given by

$$\begin{aligned}
p_{\boldsymbol{a}}(\boldsymbol{s}, \boldsymbol{s}+\boldsymbol{e}_i) &= \lambda_i, && \text{(product arrivals)}\\
p_{\boldsymbol{a}}(\boldsymbol{s}, \boldsymbol{s}-\boldsymbol{e}_i) &= \mu_i w_i \mathbb{1}_{\{x_i>0\}}, && \text{(product services)}\\
p_{\boldsymbol{a}}(\boldsymbol{s}, \boldsymbol{s}-\boldsymbol{e}_{i+2}) &= \sigma_i w_i, && \text{(machine breakdowns)}\\
p_{\boldsymbol{a}}(\boldsymbol{s}, \boldsymbol{s}+\boldsymbol{e}_{i+2}) &= q_i \nu_i, && \text{(machine repairs)}\\
p_{\boldsymbol{a}}(\boldsymbol{s}, \boldsymbol{s}) &= 1 - \lambda_i - w_i(\mu_i \mathbb{1}_{\{x_i>0\}} + \sigma_i) - q_i \nu_i. && \text{(uniformisation)}
\end{aligned}$$

All other transition probabilities are equal to zero. The tuple $(\mathscr{S}, \{\mathscr{A}_{\boldsymbol{s}} : \boldsymbol{s} \in \mathscr{S}\}, p, c)$ now fully defines the Markov decision problem at hand.

Define a deterministic policy π^* as a function from $\mathscr{S}$ to $\bigcap_{\boldsymbol{s}\in\mathscr{S}} \mathscr{A}_{\boldsymbol{s}}$ such that $\pi^*(\boldsymbol{s}) \in \mathscr{A}_{\boldsymbol{s}}$ for all $\boldsymbol{s} \in \mathscr{S}$. Let $\{\boldsymbol{X}^*(t), t \geq 0\}$ be its corresponding continuous-time Markov chain taking values in $\mathscr{S}$, which describes the state of the system over time when the repairman adheres to policy π^*. Furthermore, let

$$u^*(\boldsymbol{s}, t) = \mathbb{E}\left[\int_{z=0}^{t} c(\boldsymbol{X}^*(z), \pi^*(\boldsymbol{X}^*(z)))\, dz \mid \boldsymbol{X}^*(0) = \boldsymbol{s}\right]$$

denote the total expected costs up to time t when the system starts in state $\boldsymbol{s}$ under policy π^*.

We call the policy π^* *stable* when the average costs $g^* = \lim_{t\to\infty} \frac{u^*(\boldsymbol{s},t)}{t}$ per time unit that arise when the repairman adheres to this policy remain finite. From this, it follows that the Markov chain corresponding to the model under consideration in combination with a stable policy has a single positive recurrent class. As a result, the number g^* is independent of the initial state $\boldsymbol{s}$. Due to the definition of the cost function, the average expected costs may also be interpreted as the long-term average sum of queue lengths under policy π^*, weighted by the constants c_1 and c_2. A stable policy thus coincides with a policy for which the average number of customers in each of the queues is finite. Observe that there does not necessarily exist a stable policy for every instance of this model. In fact, a necessary (but not sufficient) condition for the existence of a stable policy reads

$$\lambda_1 < \mu_1 \frac{\nu_1}{\sigma_1 + \nu_1} \text{ and } \lambda_2 < \mu_2 \frac{\nu_2}{\sigma_2 + \nu_2}. \tag{6.1}$$

This condition implies that for each first-layer queue Q_i, the arrival rate λ_i of products is smaller than the rate at which the corresponding machine M_i is capable of processing products, given that M_i is always repaired instantly at full capacity when it breaks down. This assumption can in some sense be seen as the best-case scenario from the point of view of M_i. The latter processing rate is of course equal to the service rate μ_i times the fraction $(1/\sigma_i)/(1/\sigma_i + 1/\nu_i) = \nu_i/(\sigma_i + \nu_i)$ of time that M_i is operational under this best-case assumption. When this condition is not satisfied, there is at least one queue where

on average more products arrive per time unit than the machine can handle under any repair policy. The costs incurred will then grow without bound over time for any policy in such case, eliminating the existence of a stable policy. Thus, the converse of the necessary conditions for existence of a stable policy stated in (6.1) constitutes two different sufficient conditions for non-existence. That is, if $\lambda_1 \geq \mu_1 \frac{\nu_1}{\sigma_1+\nu_1}$ or if $\lambda_2 \geq \mu_2 \frac{\nu_2}{\sigma_2+\nu_2}$, it is guaranteed that no stable policies exist.

Any policy π^* can be characterised through its relative value function $V^*(\boldsymbol{s})$. This function is a real-valued function defined on the state space $\mathscr{S}$ given by

$$V^*(\boldsymbol{s}) = \lim_{t\to\infty}(u^*(\boldsymbol{s},t) - u^*(\boldsymbol{s}_{\text{ref}},t))$$

and represents the asymptotic difference in expected total costs incurred when starting the process in state $\boldsymbol{s}$ instead of some reference state $\boldsymbol{s}_{\text{ref}}$ (see e.g. [121, Equation (5.6.2)]). Among all policies, the optimal policy π^{opt} with relative value function V^{opt} minimises the average costs (i.e. the long-term average weighted sum of queue lengths), thus $g^{opt} = \min_{\pi^*} g^*$. Its corresponding long-term optimal actions are a solution of the Bellman optimality equations $g^{opt} + V^{opt}(\boldsymbol{s}) = \min_{\boldsymbol{a}\in\mathscr{A}_s}\{c(\boldsymbol{s},\boldsymbol{a}) + \sum_{\boldsymbol{t}\in\mathscr{S}} p_{\boldsymbol{a}}(\boldsymbol{s},\boldsymbol{t})V^{opt}(\boldsymbol{t})\}$ for all $\boldsymbol{s}\in\mathscr{S}$. For our problem, these equations are given by

$$g^{opt} + V^{opt}(x_1,x_2,w_1,w_2) = H^{opt}(x_1,x_2,w_1,w_2) + K^{opt}(x_1,x_2,w_1,w_2)$$

for every $(x_1,x_2,w_1,w_2)\in\mathscr{S}$, where H^{opt} and K^{opt} are defined in the following way. For an arbitrary policy π^* with a relative value function V^*, the function H^* is given by

$$\begin{aligned}
H^*(x_1,x_2,w_1,w_2) = {} & c_1x_1 + c_2x_2 \\
& + \lambda_1 V^*(x_1+1,x_2,w_1,w_2) + \lambda_2 V^*(x_1,x_2+1,w_1,w_2) \\
& + \mu_1 w_1 V^*((x_1-1)^+,x_2,1,w_2) + \mu_2 w_2 V^*(x_1,(x_2-1)^+,w_1,1) \\
& + \sigma_1 w_1 V^*(x_1,x_2,0,w_2) + \sigma_2 w_2 V^*(x_1,x_2,w_1,0) \\
& + \left(1 - \sum_{i=1}^{2}(\lambda_i + w_i(\mu_i+\sigma_i))\right) V^*(x_1,x_2,w_1,w_2),
\end{aligned} \tag{6.2}$$

and it models the costs and the action-independent events of product arrivals, product service completions, machine breakdowns and dummy transitions, respectively. The function K^* given by

$$\begin{aligned}
& K^*(x_1,x_2,w_1,w_2) \\
& = \min_{(q_1,q_2)\in\mathscr{A}_{(x_1,x_2,w_1,w_2)}} \{q_1\nu_1(V^*(x_1,x_2,1,w_2) - V^*(x_1,x_2,0,w_2)) \\
& \qquad + q_2\nu_2(V^*(x_1,x_2,w_1,1) - V^*(x_1,x_2,w_1,0))\}
\end{aligned} \tag{6.3}$$

models the optimal state-specific decisions of how to allocate the repair capacity over the machines and includes corrections for the uniformisation term.

As already mentioned in Section 6.1, these equations are exceptionally hard to solve analytically. Alternatively, the optimal actions can be obtained numerically by recursively defining $V^{n+1}(\boldsymbol{s}) = H^n(\boldsymbol{s}) + K^n(\boldsymbol{s})$ for an arbitrary function V^0. For $n\to\infty$, the minimising actions converge to the optimal ones (see [163] for conditions on existence and convergence). We use this procedure called *value iteration* or *successive approximation* for our numerical experiments in Section 6.6.

6.3 Structural properties of the optimal policy

As mentioned before, it is hard to give a complete, explicit characterisation of the optimal policy for the problem sketched in Section 6.2. Therefore, we derive a near-optimal policy later in Section 6.5. Nevertheless, several important structural properties of the optimal policy can be obtained. It turns out that the optimal policy is a non-idling policy, always dictates the repairman to work on one machine only and can be classified as a threshold policy. In this section, we inspect these properties more closely.

6.3.1 Non-idling property

We show in this section that the optimal policy is a non-idling policy, which means the repairman always repairs at full capacity whenever a machine is not operational, i.e. $q_1 + q_2 = 1 - w_1 w_2$. Intuitively, this makes sense, as there are no costs involved in the repairman's service. On the other hand, having less repair capacity go unused has decreasing effects on the long-term weighted number of products in the system. There is no trade-off present, and therefore the repair capacity should be used exhaustively whenever there is a machine in need of repair.

This property can be proved rigorously. Note that the minimisers of the right-hand side of (6.3) represent the optimal actions. From this, it follows that the optimal action satisfies $q_1 + q_2 = 1 - w_1 w_2$ for every state $\boldsymbol{s} \in \mathscr{S}$ (i.e. the optimal policy satisfies the non-idling property) if both $V^{opt}(x_1, x_2, 0, w_2) - V^{opt}(x_1, x_2, 1, w_2)$ and $V^{opt}(x_1, x_2, w_1, 0) - V^{opt}(x_1, x_2, w_1, 1)$ are non-negative for all $(x_1, x_2, w_1, w_2) \in \mathscr{S}$. The next proposition proves the latter condition. For the sake of reduction of the proof's complexity, it also concerns the trivial fact that under the optimal policy, the system incurs higher costs whenever the number of products in the system is increasing (i.e. $V^{opt}(x_1+1, x_2, w_1, w_2) - V^{opt}(x_1, x_2, w_1, w_2)$ and $V^{opt}(x_1, x_2 + 1, w_1, w_2) - V^{opt}(x_1, x_2, w_1, w_2)$ are non-negative).

PROPOSITION 6.3.1. *The relative value function $V^{opt}(\boldsymbol{s})$ corresponding to the optimal policy satisfies the following properties for all $\boldsymbol{s} \in \mathscr{S}$:*

1. $V^*(x_1, x_2, 0, w_2) - V^*(x_1, x_2, 1, w_2) \geq 0$ *and* $V^*(x_1, x_2, w_1, 0) - V^*(x_1, x_2, w_1, 1) \geq 0$,
2. $V^*(x_1 + 1, x_2, w_1, w_2) - V^*(x_1, x_2, w_1, w_2) \geq 0$ *and* $V^*(x_1, x_2 + 1, w_1, w_2) - V^*(x_1, x_2, w_1, w_2) \geq 0$.

PROOF. See Appendix 6.A. □

By proving that V^{opt} satisfies property 1 as stated in Proposition 6.3.1, we have established that the optimal policy is a non-idling policy, implying that $q_1 + q_2 = 1 - w_1 w_2$ at all times. We finish this section by pointing out that it is always optimal for the repairman to focus all his attention on one machine. That is, at all times, $(q_1, q_2) = (1 - w_1, 0)$ or $(q_1, q_2) = (0, 1 - w_2)$ constitutes an optimal action. This is easily derived from (6.3) in combination with property 1 in Proposition 6.3.1. Even when there are states for which $w_1 w_2 = 0$ and $\nu_1(V^*(x_1, x_2, 1, w_2) - V^*(x_1, x_2, 0, w_2)) = \nu_2(V^*(x_1, x_2, w_1, 1) - V^*(x_1, x_2, w_1, 0))$, the actions $(q_1, q_2) = (1 - w_1, 0)$ and $(q_1, q_2) = (0, 1 - w_2)$ will be optimal (although they are not uniquely optimal), so that there are always optimal policies

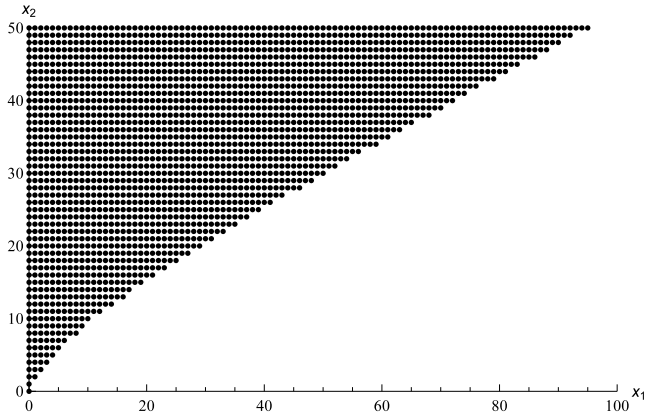

FIGURE 6.1: The optimal actions for the model instance studied in Section 6.3.2.

that concentrate all repair capacity on one machine. Therefore, K^{opt} can be simplified to

$$\begin{aligned} K^{opt}(x_1,x_2,w_1,w_2) = & \\ \min_{(q_1,q_2)\in\{(1-w_1,0),(0,1-w_2)\}} & \{q_1\nu_1(V^{opt}(x_1,x_2,1,w_2)-V^{opt}(x_1,x_2,0,w_2)) \\ & + q_2\nu_2(V^{opt}(x_1,x_2,w_1,1)-V^{opt}(x_1,x_2,w_1,0))\}. \end{aligned} \tag{6.4}$$

This is a welcome simplification when one wants to evaluate the optimal policy numerically, since now the minimum operator only involves two arguments.

6.3.2 Threshold policy

Now that we know that the optimal policy is a non-idling policy and always dictates the repairman to focus his attention on a single machine, the question arises which machine this should be. In the event both machines are down, this question is hard to answer explicitly, since the relative value function V^{opt} pertaining to the optimal policy defies an exact analysis. However, by inspection of numerical results, one can derive a partial answer.

To this end, we numerically examine the model with the settings $c_1 = c_2 = \mu_2 = \sigma_1 = \nu_1 = 1.0$, $\lambda_1 = 0.1$, $\lambda_2 = 0.2$ and $\mu_1 = \sigma_2 = \nu_2 = 0.5$. By using the simplified version (6.4) of K^{opt} in the value iteration algorithm, we numerically obtain the optimal actions for the states $(x_1,x_2,0,0)$, $x_1 \in \{0,\ldots,50\}$, $x_2 \in \{0,\ldots,100\}$. Figure 6.1 shows the optimal actions in the form of a scatter plot. Given that both machines are down, a marked point (x_1,x_2) in the scatter plot indicates that it is optimal for the repairman to repair M_2. If a certain point (x_1,x_2) is not marked, then the optimal action is to repair M_1 at full capacity.

It is suggested by Figure 6.1 that the optimal policy falls in the class of threshold policies. That is, if the optimal action for the state $(x_1,x_2,0,0)$ is to repair M_1 at full capacity, then this is also the optimal action for the states $(x_1+k,x_2,0,0)$, $k \in \mathbb{N}$. Meanwhile, if it would be optimal to repair M_2 when the system is in the state $(x_1,x_2,0,0)$,

then the optimal policy also prescribes to repair M_2 if there are fewer products waiting in Q_1, i.e. in the states $(x_1-k, x_2, 0, 0)$, $k \in \{1, \ldots, x_1\}$. Thus, for any value of x_2, the number of products in Q_1 from which the optimal policy starts taking the decision to repair M_1 can be seen as a threshold. Similar effects and definitions apply for varying numbers of products in Q_2. The figure clearly exposes a curve that marks the thresholds. At first glance, this threshold curve may seem linear. However, especially near the origin, this is not quite true.

One can reason intuitively that for any instance of the model, the optimal policy is a threshold policy. This is easily understood by the notion that an increasing number of products in Q_1 makes it more attractive for the repairman to repair M_1. Then, if it was already optimal to repair M_1, this obviously will not change. Similar notions exist for a decreasing number of products in Q_1 and varying numbers of products in Q_2. Although the threshold effects are easily understood, they are hard to prove rigorously. A possible approach to this would be to show that the difference between the arguments in (6.4) is increasing in x_1 using the same techniques as used in the proof of Proposition 6.3.1. However, this turns out to be highly challenging.

6.4 Relative value functions

Recall that for any policy π^*, we defined V^* and g^* to be its corresponding relative value function and long-run expected weighted number of products in the system, respectively. The main reason why it is hard to obtain the optimal policy π^{opt} other than through numerical means, is that its corresponding relative value function V^{opt} does not easily allow for an exact analysis. As an intermediate step, we therefore study the relative value functions of two other policies for which explicit expressions can be obtained. In Section 6.5, these two policies and their relative value functions act as a basis for the one-step policy improvement method to obtain nearly optimal heuristic policies. We first examine the static policy in Section 6.4.1, where each machine is assigned a fixed part of the repair capacity regardless of the state of the system. However, there exist instances of the model for which no static policies are available that result in a finite average cost, while stable policies are available in general. Since a one-step policy improvement approach cannot be based on a static policy in that case, we will also study the priority policy in Section 6.4.2, which dictates the repairman to prioritise a specific machine (the high-priority machine) in case both machines are not operational; i.e. in such a case, all repair capacity is then given to the high-priority machine.

6.4.1 Static policy

As the name of the static policy suggests, the actions taken under this policy do not depend on the state the system is in. Under the static policy, the repairman always has a fraction $p \in (0,1)$ of his repair capacity reserved for the repair of M_1 regardless of whether M_1 (or M_2) is down or not. Likewise, the remaining fraction $(1-p)$ is reserved for M_2. Therefore, repair on M_1 at rate $p\nu_1$ starts instantly the moment it breaks down, and the same holds for M_2 at rate $(1-p)\nu_2$. Thus, under this policy, the repairman always takes the action $(p(1-w_1), (1-p)(1-w_2))$. In the sequel, we will refer to p as the splitting parameter. It is evident that this policy is not optimal, since the repairman does not use his repair capacity exhaustively when exactly one of the two machines is down; i.e. the

static policy does not satisfy the non-idling property studied in Section 6.3.1. However, when the splitting parameter is chosen properly, this policy is not totally unreasonable either. When analysing this policy, we assume that the system is stable when adhering to it. That is, for each queue, the rate of arriving products is smaller than the rate at which the corresponding machine is capable of serving products:

$$\lambda_1 < \mu_1 \frac{p\nu_1}{\sigma_1 + p\nu_1} \text{ and } \lambda_2 < \mu_2 \frac{(1-p)\nu_2}{\sigma_2 + (1-p)\nu_2}, \tag{6.5}$$

where the two fractions denote the fractions of time M_1 and M_2 are operational, respectively.

Observe that the capacity that M_1 receives from the repairman is now completely independent of that received by M_2 at any given time and vice versa. Analysis of the relative value function of the static policy is tractable, since the machines do not compete for repair resources anymore under this policy, making the queue lengths in each of the queues uncorrelated. In a way, it is as if each machine has its own repairman now, who repairs at rate $p\nu_1$ and $(1-p)\nu_2$, respectively. Therefore, the system can be decomposed into two components which do not interact. Each of these components can be modelled as a single-server queue of M/M/1 type with server vacations occurring independently of the amount of work present in the queue. Because of this decomposition, the relative value function $V^{sta}(x_1, x_2, w_1, w_2)$ of the total system can be seen as the weighted sum of the relative value functions $V_1^{com}(x_1, w_1)$ and $V_2^{com}(x_2, w_2)$ corresponding to the two components. As a result, the long-term average cost g^{sta} is also a weighted sum of the average costs g_1^{com} and g_2^{com}:

$$\begin{aligned} &g^{sta} = c_1 g_1^{com} + c_2 g_2^{com} \text{ and} \\ &V^{sta}(x_1, x_2, w_1, w_2) = c_1 V_1^{com}(x_1, w_1) + c_2 V_2^{com}(x_2, w_2). \end{aligned} \tag{6.6}$$

To derive g_1^{com}, g_2^{com}, $V_1^{com}(x_1, w_1)$ and $V_2^{com}(x_2, w_2)$, we focus on the relative value function corresponding to one component in Section 6.4.1.1. We then finalise the analysis of V^{sta} in Section 6.4.1.2.

6.4.1.1 Relative value function for the components

We now derive the relative value function of one component of the model under the static policy and omit all indices of the parameters. Thus, we regard a single-server queue of M/M/1 type, in which products arrive at rate λ and are processed at rate μ if the machine is up. Independently of this process, the server takes a vacation after an exponentially (σ) distributed amount of time, even when there is a product in service. The service of the product is then interrupted and resumed once the server ends its vacation. A vacation takes an exponentially (ν) distributed amount of time, after which the server will process products again until the next vacation. This system can be interpreted as a Markov reward chain with states $(x, w) \in \mathscr{S}^{com}$ representing the number x of products present in the system and the state of the server being in a vacation ($w = 0$) or not ($w = 1$), where $\mathscr{S}^{com} = \mathbb{N} \times \{0, 1\}$ is its state space. The system is said to incur costs at rate $c(x, w) = x$ per time unit. After uniformisation at rate one, the transition probabilities $p^{com}(\boldsymbol{s}, \boldsymbol{t})$ from

a state $\boldsymbol{s} \in \mathscr{S}^{com}$ to a state $\boldsymbol{t} \in \mathscr{S}^{com}$ are given by

$$\begin{array}{ll} p^{com}((x,w),(x+1,w)) = \lambda, & p^{com}((x,w),(x-1,w)) = \mu w \mathbb{1}_{\{x>0\}}, \\ p^{com}((x,1),(x,0)) = \sigma, & p^{com}((x,0),(x,1)) = \nu \text{ and} \\ p^{com}((x,w),(x,w)) = (1-\lambda-w(\mu\mathbb{1}_{\{x>0\}}+\sigma)+\nu(1-w)). & \end{array}$$

All other transition probabilities are equal to zero. By this description, the Poisson equations for this Markov reward chain with long-term average costs per time unit g^{com} and relative value function $V^{com}(x,w)$ are given by

$$\begin{aligned} g^{com} + V^{com}(x,w) = x &+ \lambda V^{com}(x+1,w) + \mu w V^{com}((x-1)^+,w) \\ &+ \sigma w V^{com}(x,0) + \nu(1-w)V^{com}(x,1) \\ &+ (1-\lambda-w(\mu+\sigma)-\nu(1-w))V^{com}(x,w) \end{aligned} \tag{6.7}$$

for all $(x,w) \in \mathbb{N} \times \{0,1\}$.

To solve these equations, we first observe that the completion time for a product from the moment its service is started until it leaves the system consists of an exponentially (μ) distributed amount of actual service time and possibly some interruption time due to server vacations. When interruption takes place, the number of interruptions is geometrically $(\frac{\mu}{\mu+\sigma})$ distributed due to the Markovian nature of the model. Combined with the fact that every interruption takes an exponential (ν) amount of time, this means that the total interruption time, given that it is positive, is exponentially $(\frac{\mu\nu}{\mu+\sigma})$ distributed. Thus, the completion time consists of an exponential (μ) service phase and also, with a probability $\frac{\sigma}{\mu+\sigma}$ that there is at least one interruption, an exponential $(\frac{\mu\nu}{\mu+\sigma})$ interruption phase. The above implies that the distribution of the completion time falls in the class of Coxian distributions with two states. Due to this observation, the average costs per time unit g^{com} incurred by a component can be calculated by the use of standard queueing theory; see Remark 6.4.2. However, we are also interested in the relative value function of the component. If the server would only start a vacation if there is at least one product in the queue, the component could in principle be modelled as an M/Cox(2)/1 queue by incorporating the interruption times into the service times (i.e. by replacing the service times with the completion times). For the M/Cox(2)/1 queue, it is known that the relative value function can be expressed as a second-order polynomial in the queue length (cf. [35]). However, in our case, a server may also start a vacation during an idle period, so that products arriving at an empty system may not be served instantly. Nevertheless, it is reasonable to conjecture that the relative value function V^{com} is a second-order polynomial too.

If this conjecture holds, substituting $V^{com}(x,0) = \alpha_1 x^2 + \alpha_2 x + \alpha_3$ and $V^{com}(x,1) = \beta_1 x^2 + \beta_2 x + \beta_3$ in (6.7) should lead to a consistent system of equations and give a solution for the coefficients. After substitution, we find the equations

$$\begin{aligned} g^{com} + \alpha_3 &= \lambda(\alpha_1+\alpha_2) + (1-\nu)\alpha_3 + \nu\beta_3, \\ g^{com} + \beta_3 &= \sigma\alpha_3 + \lambda(\beta_1+\beta_2) + (1-\sigma)\beta_3, \\ g^{com} + \alpha_1 x^2 + \alpha_2 x + \alpha_3 &= ((1-\nu)\alpha_1 + \nu\beta_1)x^2 + (1 + 2\lambda\alpha_1 + (1-\nu)\alpha_2 + \nu\beta_2)x \\ &\quad + \lambda(\alpha_1+\alpha_2) + (1-\nu)\alpha_3 + \nu\beta_3, \\ g^{com} + \beta_1 x^2 + \beta_2 x + \beta_3 &= (\sigma\alpha_1 + (1-\sigma)\beta_1)x^2 + \big(1 + \sigma\alpha_2 + 2(\lambda-\mu)\beta_1 \\ &\quad + (1-\sigma)\beta_2\big)x + \sigma\alpha_3 + (\lambda+\mu)\beta_1 + (\lambda-\mu)\beta_2 + (1-\sigma)\beta_3 \end{aligned}$$

for all $x \in \mathbb{N}_+$. One can easily verify that the system of equations is indeed consistent. By solving for the coefficients, a solution for g^{com} and V^{com} up to a constant can be found. The constant can be chosen arbitrarily (e.g. by assuming that $V^{com}(0,1) = 0$), but is of no importance. In principle, there may exist other solutions to (6.7) that do not behave like a second-order polynomial in x. In fact, when the state space is not finite, as is the case in our model, it is known that there are many pairs of g and V that satisfy the Poisson equations (6.7) (see e.g. [37]). There is only one pair satisfying $V(0,1) = 0$ that is the correct stable solution, however, and we refer to this as the unique solution. Showing that a solution to (6.7) is the unique solution involves the construction of a weighted norm so that the Markov chain is geometrically recurrent with respect to that norm. This weighted norm imposes extra conditions on the solution to the Poisson equations, so that the unique solution can be identified. The next lemma summarises the solution resulting from the set of equations above and states that this is also the unique solution.

LEMMA 6.4.1. *For a stable component instance, the long-term average number of products g^{com} and the relative value function V^{com} are given by*

$$g^{com} = \frac{\lambda((\sigma+\nu)^2 + \mu\sigma)}{(\mu\nu - \lambda(\sigma+\nu))(\sigma+\nu)}, V^{com}(x,0) = \alpha_1 x^2 + \alpha_2 x + \alpha_3$$
$$\text{and } V^{com}(x,1) = \alpha_1 x^2 + \alpha_1 x, \tag{6.8}$$

where

$$\alpha_1 = \frac{\sigma+\nu}{2(\mu\nu - \lambda(\sigma+\nu))}, \alpha_2 = \frac{2\mu+\sigma+\nu}{2(\mu\nu-\lambda(\sigma+\nu))} \text{ and } \alpha_3 = \frac{\lambda\mu}{(\mu\nu-\lambda(\sigma+\nu))(\sigma+\nu)},$$

when taking $V^{com}(0,1) = 0$ as a reference value.

PROOF. One simply verifies by substitution that the solution given in (6.8) satisfies the Poisson equations in (6.7) and $V^{com}(0,1) = 0$. It is left to show that the above solution is the unique solution. To this end, we use [37, Theorem 6]. Suppose that there exists a finite subset of states M and a weight function $u : \mathscr{S}^{com} \to \{0,1\}$ such that the Markov chain, which satisfies the stability and aperiodicity conditions needed for the theorem to hold, is u-geometrically recurrent, i.e.

$$R_{M,u}(x,w) = \sum_{(x',w') \notin M} \frac{p^{com}((x,w),(x',w'))u(x',w')}{u(x,w)} < 1$$

for all $(x,w) \in \mathscr{S}$ and

$$||c||_u = \sup_{s \in \mathscr{S}^{com}} \frac{|c(s)|}{u(s)} < \infty.$$

Then, this theorem implies that a pair (g,V) satisfying the Poisson equations (6.7) is the unique solution when

$$||V||_u = \sup_{s \in \mathscr{S}^{com}} \frac{|V(s)|}{u(s)} < \infty.$$

To invoke this theorem, we set $M = \{(0,0),(0,1)\}$ and $u(x,w) = (1+\delta)^x(1-\epsilon)^w$ for any

$$\delta \in \left(0, \frac{\mu+\nu+\sigma-\sqrt{(\lambda-\mu-\nu-\sigma)^2 + 4\,(\lambda\nu-\mu\nu+\lambda\sigma)}}{2\lambda} - \frac{1}{2}\right)$$

and for any

$$\epsilon \in \left(\frac{\lambda}{\nu}\delta, \frac{\delta\mu - \lambda\delta(1+\delta)}{\delta\mu - \lambda\delta(1+\delta) + \sigma(1+\delta)} \right).$$

Then, we have that

$$\begin{aligned} R_{M,u}(x,w) = \lambda(1+\delta) + w\left(\mu \mathbb{1}_{\{x>1\}} \frac{1}{1+\delta} + \sigma \frac{1}{1-\epsilon}\right) \\ + \nu(1-w)(1-\epsilon) + (1-\lambda - w(\mathbb{1}_{\{x>1\}}\mu + \sigma) - (1-w)\nu). \end{aligned}$$

For all $x \in \mathbb{N}$, the lower bound on ϵ ensures that $R_{M,u}(x,0) < 1$, and the upper bound guarantees that $R_{M,u}(x,1) < 1$. The upper bound of δ is derived by equating the two bounds of ϵ and thus warrants that the lower bound of ϵ does not exceed the upper bound of ϵ. In turn, the stability condition $\lambda < \mu\frac{\nu}{\sigma+\nu}$ (see (6.5)) guarantees that the upper bound of δ is positive. Observe that for the assessment of the validity of the conditions $||c||_u < \infty$ and $||V^{com}||_u < \infty$, the value of w does not play an essential role, as it can only influence the value of $u(x,w)$ up to a finite factor $(1-\epsilon)$ for any $x \in \mathbb{N}$. We clearly have that the cost function $c(x,w) = x$ satisfies $||c||_u < \infty$, since it is linear in x and the weight function u is exponential in x. Likewise, the function V^{com} as given in (6.8) satisfies $||V^{com}||_u < \infty$, since it is a quadratic polynomial in x, whereas $u(x,w)$ behaves exponentially in x. Hence, by [37, Theorem 6], the solution given by (6.8) is the unique solution to the Poisson equations. □

This concludes the derivation of the relative value function for a component with parameters λ, μ, σ and ν.

REMARK 6.4.1. For $\sigma = 0$ and $w = 1$, the component model degenerates to a regular M/M/1 queue. As expected, g^{com} and $V^{com}(x,1)$ then simplify to the well-known expressions $g^{M/M/1} = \frac{\lambda}{\mu-\lambda}$ and $V^{M/M/1}(x) = \frac{1}{2(\mu-\lambda)}x(x+1)$. For the general case, we may rewrite $V^{com}(x,1) = \frac{1}{2(\mu\frac{\nu}{\sigma+\nu}-\lambda)}x(x+1)$. Observe that $\mu\frac{\nu}{\sigma+\nu}$ is the maximum rate at which the server is able to process products in the long term. When interpreting this as an effective service rate, we may conclude that the structure of the relative value function V^{com} is similar to that of the regular M/M/1 queue.

REMARK 6.4.2. As observed above, a component can alternatively be modelled as a single-server vacation queue with the Coxian completion time of a product regarded as the service time and with server vacations occurring exclusively when the queue is empty. As a result, the average costs per time unit, or rather, the average queue length g^{com} (including any possible product in service) can also be obtained by applying the Fuhrmann-Cooper decomposition (cf. [102]) similarly to the computations that led to (4.6) in Section 4.4.

6.4.1.2 Resulting expression for V^{sta}

We now turn back to the relative value function of the complete model as described in Section 6.2 under the static policy with parameter p. As mentioned before, this model consists of two components with rates $\lambda_1, \mu_1, \sigma_1, p\,\nu_1$ and $\lambda_2, \mu_2, \sigma_2, (1-p)\nu_2$, respectively. Now that we have found an expression for the relative value functions pertaining to one such component, we readily obtain an expression for the relative value function for the complete system. Combining (6.6) with Lemma 6.4.1 results in the following theorem.

THEOREM 6.4.2. *Given that the stability conditions in* (6.5) *are satisfied, the long-term average costs g_p^{sta} and the relative value function $V_p^{sta}(x_1,x_2,w_1,w_2)$ corresponding to the static policy with parameter p are given by*

$$g_p^{sta} = c_1\frac{\lambda_1((\sigma_1+p\nu_1)^2+\mu_1\sigma_1)}{(\sigma+p\nu_1)(\mu_1 p\nu_1-\lambda_1(\sigma_1+p\nu_1))} + c_2\frac{\lambda_2((\sigma_2+(1-p)\nu_2)^2+\mu_2\sigma_2)}{(\sigma_2+(1-p)\nu_2)(\mu_2(1-p)\nu_2-\lambda_2(\sigma_2+(1-p)\nu_2))}$$

and

$$V_p^{sta}(x_1,x_2,w_1,w_2) = \alpha_{1,1}c_1x_1^2+c_1(\alpha_{2,1}(1-w_1)+\alpha_{1,1}w_1)x_1+\alpha_{3,1}c_1(1-w_1) + \alpha_{1,2}c_2x_2^2+c_2(\alpha_{2,2}(1-w_2)+\alpha_{1,2}w_2)x_2+\alpha_{3,2}c_2(1-w_2)$$

for all $(x_1,x_2,w_1,w_2)\in\mathscr{S}$, where

$$\alpha_{1,1}=\frac{\sigma_1+p\nu_1}{2\mu_1 p\nu_1-\lambda_1(\sigma_1+p\nu_1)},\alpha_{1,2}=\frac{\sigma_2+(1-p)\nu_2}{2\mu_2(1-p)\nu_2-\lambda_2(\sigma_2+(1-p)\nu_2)},$$
$$\alpha_{2,1}=\frac{2\mu_1+\sigma_1+p\nu_1}{2\mu_1 p\nu_1-\lambda_1(\sigma_1+p\nu_1)},\alpha_{2,2}=\frac{2\mu_2+\sigma_2+(1-p)\nu_2}{2\mu_2(1-p)\nu_2-\lambda_2(\sigma_2+(1-p)\nu_2)},$$
$$\alpha_{3,1}=\frac{\lambda_1\mu_1}{(\mu_1 p\nu_1-\lambda_1(\sigma_1+p\nu_1))(\sigma_1+p\nu_1)}\ \textit{and}$$
$$\alpha_{3,2}=\frac{\lambda_2\mu_2}{(\mu_2(1-p)\nu_2-\lambda_2(\sigma_2+(1-p)\nu_2))(\sigma_2+(1-p)\nu_2)}.$$

6.4.2 Priority policy

In the previous section, we have derived an explicit expression for the relative value function for the static policy. In Section 6.5, this policy will act as an initial policy for the one-step policy improvement algorithm to obtain a well-performing heuristic policy. However, for certain instances of the model, there may be no static policy available for which the system is stable, whereas the optimal policy does result in stable queues. When this happens, one-step policy improvement based on the static policy is not feasible, since the initial policy for this procedure must result in a stable system. In these cases, a priority policy may still result in stability and thus be suitable as an initial policy, so that a heuristic policy can still be obtained. For this reason, we study the relative value function of the priority policy in the current section.

Under priority policy π_i^{prio}, the repairman always prioritises the repair of machine M_i, which we will refer to as the high-priority machine. This means that in case both machines are down, the repairman allocates his full capacity to M_i as a high-priority machine. If there is only one machine unoperational, the repairman dedicates his capacity to the broken machine regardless of whether it is the high-priority machine. In case all machines are operational, the repairman obviously remains idle. Thus, the repairman always takes the action $((1-w_1),w_1(1-w_2))$ if $i=1$ or $((1-w_1)w_2,(1-w_2))$ if $i=2$. The priority policy π_1^{prio}, where M_1 acts as the high-priority machine, is stable if and only if for each queue the rate at which products arrive is smaller than the effective service rate of its

machine:

$$\lambda_1 < \mu_1 \frac{\nu_1}{\sigma_1 + \nu_1} \text{ and } \lambda_2 < \mu_2^{eff}, \tag{6.9}$$

where μ_2^{eff} refers to the effective service rate of M_2. The right-hand side of the first inequality represents the effective service rate of the high-priority machine M_1 and consists of the actual service rate μ_1 times the fraction of time M_1 is operational under the priority policy. The effective service rate of M_2 analogously satisfies

$$\mu_2^{eff} = \mu_2 \frac{\frac{1}{\sigma_2}}{\frac{1}{\sigma_2} + \frac{\xi}{\nu_1} + \mathbb{E}[Z_2]}. \tag{6.10}$$

The expression $\frac{\xi}{\nu_1} + \mathbb{E}[Z_2]$ in the right-hand side represents the expected downtime of M_2. The constant ξ refers to the probability that M_2 observes the repairman busy on M_1 when it breaks down, so that $\frac{\xi}{\nu_1}$ represents the expected time M_2 has to wait after its breakdown until the start of its repair as a result of an M_1 failure. The probability ξ is computed by the fixed-point equation

$$\xi = \frac{\sigma_1}{\sigma_1 + \sigma_2}\left(\frac{\sigma_2}{\nu_1 + \sigma_2} + \frac{\nu_1}{\nu_1 + \sigma_2}\xi\right),$$

which leads to

$$\xi = \frac{\sigma_1}{\sigma_1 + \sigma_2 + \nu_1}. \tag{6.11}$$

Likewise, $\mathbb{E}[Z_2]$ represents the expected time from the moment the repairman starts repair on M_2 until its finish and is computed by the fixed-point equation

$$\mathbb{E}[Z_2] = \frac{1}{\sigma_1 + \nu_2} + \frac{\sigma_1}{\sigma_1 + \nu_2}\left(\frac{1}{\nu_1} + \mathbb{E}[Z_2]\right),$$

which leads to

$$\mathbb{E}[Z_2] = \frac{1}{\nu_2} + \frac{\sigma_1}{\nu_1 \nu_2}.$$

By repeating the arguments above, it is easy to see that the priority policy π_2^{prio} is stable if and only if

$$\lambda_1 < \mu_1^{eff} \text{ and } \lambda_2 < \mu_2 \frac{\nu_2}{\sigma_2 + \nu_2}, \tag{6.12}$$

where μ_1^{eff} has an expression similar to μ_2^{eff}, but with indices interchanged.

In the remainder of this section, we study the relative function corresponding to the priority policy under the assumption that this policy is stable. We will only study the priority policy π_1^{prio} where M_1 acts as the high-priority machine. Results for the other case follow immediately by similar arguments or simply by interchanging indices. Therefore, we drop the machine-specific index in this section, so that V^{prio} actually refers to V_1^{prio}.

Deriving an expression for the relative value function V^{prio} of the priority policy is hard. Before, in the case of the static policy, the model could be decomposed into several components which exhibit no interdependence. This allowed us to obtain an explicit expression for V^{sta}. In contrast, a similar decomposition under the current policy does

lead to *interacting* components. The first component, which contains the high-priority machine M_1 and its corresponding queue, acts independently of any other component, since M_1 is not affected by M_2 when accessing repair resources. However, M_2 is affected by M_1. This interference causes the second component, which contains the other machine and its queue of products, to become dependent on the events occurring in the first component. Therefore, there exist correlations, which makes an explicit analysis of V^{prio} hard. Nevertheless, we are still able to derive certain characteristics of the relative value function.

When decomposing the model in the same way as was done in Section 6.4.1, we have, similar to (6.6), that the long-term average costs g^{prio} per time unit and the relative value function V^{prio} pertaining to the priority policy can be written as

$$\begin{aligned} &g^{prio} = c_1 g^{prc} + c_2 g^{nprc} \text{ and} \\ &V^{prio}(x_1, x_2, w_1, w_2) = c_1 V^{prc}(x_1, w_1) + c_2 V^{nprc}(x_2, w_1, w_2), \end{aligned} \tag{6.13}$$

where g^{prc} and $V^{prc}(x_1, w_1)$ are the long-term average costs and the relative value function pertaining to the first component, which we will also call the priority component. Similarly, g^{nprc} and $V^{nprc}(x_2, w_1, w_2)$ denote the long-term average costs and the relative value function of the second component, which we will also refer to as the non-priority component. In both of these subsystems, the products present are each assumed to incur costs at rate one. Note that the function $V^{nprc}(x_2, w_1, w_2)$ of the second component now includes w_1 as an argument, since the costs incurred in the second component are now dependent on the state of M_1 in the first component. We first obtain an explicit expression for V^{prc}. Then, as V^{nprc} defies an explicit analysis due to the aforementioned dependence, we make several conjectures on its form in Section 6.4.2.2. In Section 6.5, it will turn out that these conjectures still allow us to use π^{prio} as an initial policy for the one-step policy improvement algorithm.

6.4.2.1 Relative value function for the priority component

In the priority component, the machine M_1 faces no competition in accessing repair facilities. If M_1 breaks down, the repairman immediately starts repairing M_1 at rate ν_1. Thus, from the point of view of M_1, it is as if M_1 has its own dedicated repairman. Therefore, the priority component behaves completely similar to a component of the static policy studied in Section 6.4.1.1, but now with $\lambda_1, \mu_1, \sigma_1$ and ν_1 as product arrival, product service, machine breakdown and machine repair rates. As a result, we obtain by Lemma 6.4.1 that, when products in the queue incur costs at rate one, the long-term average costs g^{prc} and the relative value function V^{prc} are given by

$$\begin{aligned} &g^{prc} = \frac{\lambda_1((\sigma_1+\nu_1)^2 + \mu_1\sigma_1)}{(\sigma_1+\nu_1)(\mu_1\nu_1 - \lambda_1(\sigma_1+\nu_1))}, V^{prc}(x_1, 0) = \upsilon_1 x_1^2 + \upsilon_2 x_1 + \upsilon_3 \\ &\text{and } V^{prc}(x_1, 1) = \upsilon_1 x_1^2 + \upsilon_1 x_1, \end{aligned} \tag{6.14}$$

for $x_1 \in \mathbb{N}$, where

$$\begin{aligned} &\upsilon_1 = \frac{\sigma_1+\nu_1}{2(\mu_1\nu_1 - \lambda_1(\sigma_1+\nu_1))}, \upsilon_2 = \frac{2\mu_1+\sigma_1+\nu_1}{2(\mu_1\nu_1 - \lambda_1(\sigma_1+\nu_1))} \\ &\text{and } \upsilon_3 = \frac{\lambda_1\mu_1}{(\mu_1\nu_1 - \lambda_1(\sigma_1+\nu_1))(\sigma_1+\nu_1)}, \end{aligned}$$

when taking $V^{prc}(0,1) = 0$ as a reference value.

6.4.2.2 Heuristic for approximating the relative value function for the non-priority component

As mentioned earlier, the relative value function V^{nprc} of the non-priority component defies an explicit analysis due to its dependence on the priority component. Explorative numerical experiments suggest that V^{nprc} asymptotically behaves like a second-order polynomial in x_2 as $x_2 \to \infty$. We support this insight by arguments from queueing theory, which are given in Conjecture 6.4.3 below. Building on this, we also pose certain conjectures on the first-order and second-order coefficients of this polynomial. This leads to a heuristic for approximating the relative value function for the non-priority component, which we present in this section. Finally, we present an approximation for the long-term expected costs g^{nprc}.

In the non-priority component, products arrive at rate λ_2 and are served at rate μ_2 by M_2 when it is operational. Independently of this, M_2 breaks down at rate σ_2 when it is operational. In case M_2 is down, it gets repaired at rate ν_2 if M_1 is operational and at rate zero otherwise. Obviously, if M_1 is operational, it breaks down at rate σ_1. Otherwise, it gets repaired at rate ν_1. The resulting system can again be formulated as a Markov reward chain with states $(x_2, w_1, w_2) \in \mathscr{S}^{nprc}$, representing the number of products in the component (x_2) and the indicator variables corresponding to each of the machine's operational states (w_1, w_2), where $\mathscr{S}^{nprc} \in \mathbb{N} \times \{0,1\}^2$ is its state space. This chain is said to incur costs at rate $c(x_2, w_1, w_2) = x_2$. After uniformisation at rate one, the transition probabilities $p^{nprc}(\boldsymbol{s}, \boldsymbol{t})$ from a state $\boldsymbol{s} \in \mathscr{S}^{nprc}$ to a state $\boldsymbol{t} \in \mathscr{S}^{nprc}$ are given by

$$
\begin{aligned}
&p^{nprc}((x_2,w_1,w_2),(x_2+1,w_1,w_2)) = \lambda_2, \quad p^{nprc}((x_2,w_1,w_2),(x_2-1,w_1,w_2)) \\
&\qquad\qquad = \mu_2 w_2 \mathbb{1}_{\{x_2>0\}}, \\
&p^{nprc}((x_2,1,w_2),(x_2,0,w_2)) = \sigma_1, \qquad p^{nprc}((x_2,w_1,1),(x_2,w_1,0)) = \sigma_2, \\
&p^{nprc}((x_2,0,w_2),(x_2,1,w_2)) = \nu_1, \qquad p^{nprc}((x_2,1,0),(x_2,1,1)) = \nu_2 \text{ and} \\
&p^{nprc}((x_2,w_1,w_2),(x_2,w_1,w_2)) = (1-\lambda_2-\sigma_1 w_1 - w_2(\mu_2 \mathbb{1}_{\{x_2>0\}} + \sigma_2) \\
&\qquad\qquad -\nu_1(1-w_1) - \nu_2 w_1 (1-w_2)).
\end{aligned}
$$

All other transition probabilities are equal to zero. For this Markov reward chain, the Poisson equations are given by

$$
\begin{aligned}
&g^{nprc} + V^{nprc}(x_2,w_1,w_2) \\
&= x_2 + \lambda_2 V^{nprc}(x_2+1,w_1,w_1) + \mu_2 w_2 V^{nprc}((x_2-1)^+,w_1,1) \\
&\quad + \sigma_1 w_1 V^{nprc}(x_2,0,w_2) + \sigma_2 w_2 V^{nprc}(x_2,w_1,0) \\
&\quad + \nu_1(1-w_1) V^{nprc}(x_2,1,w_2) + \nu_2 w_1 (1-w_2) V^{nprc}(x_2,1,1) \\
&\quad + (1-\lambda_2-\sigma_1 w_1 - w_2(\mu_2+\sigma_2) - \nu_1(1-w_1) - \nu_2 w_1(1-w_2)) \\
&\qquad \times V^{nprc}(x_2,w_1,w_2).
\end{aligned} \tag{6.15}
$$

CONJECTURE 6.4.3. *Assume that the stability conditions in* (6.9) *are satisfied. Then, the relative value function $V^{nprc}(x_2,w_1,w_2)$ of the non-priority component asymptotically behaves as a second-order polynomial in x_2 with second-order coefficient $\phi_1 = \frac{1}{2}(\mu_2^{eff} - \lambda_2)^{-1}$ as $x_2 \to \infty$ for each $w_1, w_2 \in \{0,1\}$, where μ_2^{eff} represents the effective service rate of M_2 as given in* (6.10).

ARGUMENT. Recall that $V^{nprc}(x_2+1,w_1,w_2)-V^{nprc}(x_2,w_1,w_2)$ represents the long-term difference in total expected costs incurred in the non-priority component when starting the system in state (x_2+1,w_1,w_2) instead of (x_2,w_1,w_2). Since every customer generates costs at rate one per time unit, it is easily seen by a sample-path comparison argument that this difference asymptotically (for $x_2 \to \infty$) amounts to the expected time it takes for the queue to empty when the system is started in the state (x_2+1,w_1,w_2). For small values of x_2, this difference may depend slightly on w_1, since the event of M_1 being down at the start of the process may have a relatively significant impact on the time to empty the queue, as the first repair of M_2 is likely to take longer than usual. However, as x_2 becomes larger, the time needed for the queue to empty becomes larger too, so that the process describing the conditions of the machines is more likely to have advanced towards an equilibrium in the meantime. As a result, the initial value of w_1 does not have a relatively significant impact on the difference (i.e. the time for the queue to empty) for larger x_2 values. In fact, the extra delay in the time to empty imposed by an initial failure of M_1 is expected to converge to a constant as x_2 increases. Based on these observations, we expect that asymptotically, the value w_1 will only appear in the first-order coefficients of $V^{nprc}(x_2,w_1,w_2)$ when regarding it as a polynomial function in x_2, but not in higher-order coefficients. This asymptotic linear effect is studied in Conjecture 6.4.4. We also expect that V^{nprc} starts to exhibit this asymptotic behaviour very quickly as x_2 increases, since the process describing the conditions of the machines regenerates each time M_2 is repaired and thus moves to an equilibrium rather quickly.

Now that we have identified the contribution of w_1, we study the behaviour of V^{nprc} in the direction of x_2 that is not explained by w_1. When ignoring the interaction with the priority queue (thus ignoring w_1), the queue of products in the non-priority component may be interpreted as an M/PH/1 queue, by incorporating the service interruptions (consisting of M_1 and M_2 repairs) into the service times of the products. Thus, queueing-theoretic intuition suggests that the relative value function for our model may behave similarly to that of the M/PH/1 queue, particularly if the degree of interdependence between the queue lengths of Q_1 and Q_2 is not very high. It is known that the relative value function of such a queue is a quadratic polynomial (see e.g. [35]). Therefore, asymptotically, V^{nprc} is likely to behave as a quadratic polynomial too. The second-order coefficient of the relative value function of the M/PH/1 queue satisfies the form $\frac{1}{2}(\mu_2^{eff}-\lambda_2)^{-1}$, where λ_2 is the arrival rate and μ_2^{eff} is the effective service rate, i.e. the maximum long-term rate at which the server can process the products. As observed in Remark 6.4.1, the second-order coefficient α_1 of the static component in Lemma 6.4.1 is also of this form, which is independent of the value of w_2. Therefore, it is reasonable to assume that the second-order coefficient of V^{nprc} also satisfies this form, although it is independent of the values w_1 and w_2. The involved effective service rate of M_2, μ_2^{eff}, is given in (6.10). By combining all arguments above, the conjecture follows. □

Note that the first-order coefficient of the polynomial, unlike the second-order coefficient, is expected to be dependent on w_1 as mentioned in the argument of Conjecture 6.4.3, but also on w_2, in line with the results on the components of the static policy. The first-order coefficient is studied in the next conjecture.

CONJECTURE 6.4.4. *Suppose that Conjecture 6.4.3 holds true, so that, asymptotically,*

$$V^{nprc}(x_2,0,0)=\phi_1x_2^2+\phi_2x_2+\phi_3,\qquad V^{nprc}(x_2,1,0)=\phi_1x_2^2+\psi_2x_2+\psi_3,$$
$$V^{nprc}(x_2,0,1)=\phi_1x_2^2+\chi_2x_2+\chi_3 \text{ and}\qquad V^{nprc}(x_2,1,1)=\phi_1x_2^2+\omega_2x_2+\omega_3 \quad (6.16)$$

as $x_2\to\infty$. *Then,*

$$\psi_2=\phi_2-\Delta_{1,0},\chi_2=\phi_2-\Delta_{0,1},\omega_2=\phi_2-\Delta_{1,1},$$

where

$$\Delta_{0,1}=\frac{\mu_2(\nu_1+\sigma_1)(\nu_1+\nu_2+\sigma_1+\sigma_2)}{\mu_2\nu_1\nu_2(\nu_1+\sigma_1+\sigma_2)-\lambda_2(\nu_1+\sigma_1)(\nu_1(\nu_2+\sigma_2)+\sigma_2(\nu_2+\sigma_1+\sigma_2))},$$
$$\Delta_{1,0}=\frac{\mu_2\nu_2(\nu_1+\sigma_1+\sigma_2)}{\mu_2\nu_1\nu_2(\nu_1+\sigma_1+\sigma_2)-\lambda_2(\nu_1+\sigma_1)(\nu_1(\nu_2+\sigma_2)+\sigma_2(\nu_2+\sigma_1+\sigma_2))}$$

and

$$\Delta_{1,1}=\frac{\mu_2(\nu_1+\nu_2+\sigma_1)(\nu_1+\sigma_1+\sigma_2)}{\mu_2\nu_1\nu_2(\nu_1+\sigma_1+\sigma_2)-\lambda_2(\nu_1+\sigma_1)(\nu_1(\nu_2+\sigma_2)+\sigma_2(\nu_2+\sigma_1+\sigma_2))}.$$

ARGUMENT. The relative value function V^{nprc} is expected to satisfy the Poisson equations given in (6.15), also asymptotically for $x_2\to\infty$. When substituting (6.16) into (6.15) for $x_2>0$, the constraints on ϕ_2, χ_2, ψ_2 and ω_2 mentioned above are necessary for the first-order terms in x_2 on both sides of the equations to be equal. □

REMARK 6.4.3. As costs in the non-priority component are generated primarily by having customers in the queue, we expect the values of ϕ_3, χ_3, ψ_3 and ω_3 in (6.16) to be of very moderate significance compared to the second-order and first-order coefficients. As mentioned before, we also expect that V^{nprc} starts to exhibit its asymptotic behaviour very quickly as x_2 increases. Although we have not found an explicit solution for the first-order coefficients ϕ_2, χ_2, ψ_2 and ω_2, we can therefore still obtain accurate approximations for expressions such as $V^{prio}(x_1,x_2,1,0)-V^{prio}(x_1,x_2,0,0)$ and $V^{prio}(x_1,x_2,0,1)-V^{prio}(x_1,x_2,0,0)$ based on the information we have obtained. In particular, by combining the results in (6.13), (6.14), Conjecture 6.4.3 and Conjecture 6.4.4, we have that

$$V^{prio}(x_1,x_2,1,0)-V^{prio}(x_1,x_2,0,0)\approx c_1((\upsilon_1-\upsilon_2)x_1-\upsilon_3)-c_2\Delta_{1,0}x_2,$$
$$V^{prio}(x_1,x_2,0,1)-V^{prio}(x_1,x_2,0,0)\approx -c_2\Delta_{0,1}x_2 \quad (6.17)$$

with the parameters υ_1, υ_2, υ_3, $\Delta_{1,0}$ and $\Delta_{0,1}$ as previously defined in this section. These two accurate approximations allow us to apply the one-step policy improvement algorithm based on the priority policy in Section 6.5.1.2.

In the two conjectures above, we have not studied the long-term expected costs per time unit g^{nprc}. However, to predict which of the two possible priority policies π_1^{prio} and π_2^{prio} will lead to the best one-step improved policy, we will need an expression for the overall long-term average costs g^{prio}, which includes the costs g^{nprc} generated by the non-priority queue. Therefore, we end this section by deriving an approximation for g^{prio}, which is obtained by combining (6.13) and (6.14) with an independence argument.

APPROXIMATION 6.4.5. *An accurate approximation for the long-term expected costs per time unit g^{prio} is given by*

$$g_{app}^{prio} \approx c_1 \frac{\lambda_1((\sigma_1+\nu_1)^2+\mu_1\sigma_1)}{(\sigma_1+\nu_1)(\mu_1\nu_1-\lambda_1(\sigma_1+\nu_1))} + c_2 g_{app}^{nprc}, \tag{6.18}$$

where

$$g_{app}^{nprc} = \lambda_2 \mathbb{E}[C_{app}] + \frac{\lambda_2^2 \mathbb{E}[C_{app}^2]}{2(1-\lambda_2\mathbb{E}[C_{app}])} + \lambda_2 \frac{\sigma_2 \mathbb{E}[D^2]}{2(1+\sigma_2 ED)}, \tag{6.19}$$

$$\mathbb{E}[C_{app}^i] = (-1)^i \frac{d^i}{ds^i}\left(\frac{\mu_2}{\mu_2+s+\sigma_2(1-\widetilde{D}(s))}\right)\Bigg|_{s=0},$$

$$\mathbb{E}[D^i] = (-1)^i \frac{d^i}{ds^i}\widetilde{D}(s)\Big|_{s=0}, \qquad \widetilde{D}(s) = \left((1-\xi)+\xi\frac{\nu_1}{\nu_1+s}\right)\frac{\nu_2}{\nu_2+s+\sigma_1(1-\frac{\nu_1}{\nu_1+s})}$$

and ξ is defined as in (6.11).

JUSTIFICATION. The form of (6.18) is a consequence of (6.13) and (6.14). It thus remains to obtain an approximation for g^{nprc}. We do this by ignoring the interaction between the two components. Inspired by Remark 6.4.2, we approximate g^{nprc} by studying the queue length in an M/G/1 queue with server vacations. As service times of this vacation queue, we take the completion times C, which incorporate the time lost due to service interruptions as a result of a breakdown of M_2 during service. The server vacations, which start each time the queue becomes empty, include the downtimes of M_2 following a breakdown occurring when the queue is empty. Let $\tilde{D}(s) = \mathbb{E}[e^{-sD}]$ be the Laplace-Stieltjes transform representing the duration D of a downtime of M_2. This period D consists of an exponential (ν_2) repair time R_2, of which the distribution is represented by the Laplace-Stieltjes transform $\widetilde{R}_2(s) = \frac{\nu_2}{\nu_2+s}$, and a Poisson ($\sigma_1 R_2$) number of interruptions N, each caused by a breakdown of M_1. Since M_1 has priority, these interruptions take an exponential (ν_1) repair time R_1, of which the distribution is represented by the Laplace-Stieltjes transform $\widetilde{R}_1(s) = \frac{\nu_1}{\nu_1+s}$. Finally, when M_2 breaks down, it will have to wait with probability ξ (as defined in (6.11)) for an M_1-repair to finish before repair on M_2 can start. Since the repair time of M_1 is memoryless, the Laplace-Stieltjes transform of the distribution of this waiting time also equals $\widetilde{R}_1(s)$. Thus, we have that

$$\begin{aligned}
\tilde{D}(s) &= \big((1-\xi)+\xi\widetilde{R}_1(s)\big)\int_{t=0}^{\infty} e^{-st}\left(\sum_{n=0}^{\infty}\widetilde{R}_1^n(s)\mathbb{P}(N=n)\right)\nu_2 e^{-\nu_2 t}dt \\
&= \big((1-\xi)+\xi\widetilde{R}_1(s)\big)\int_{t=0}^{\infty} e^{-st}\left(\sum_{n=0}^{\infty} e^{-\sigma_1 t}\frac{(\sigma_1 t\widetilde{R}_1(s))^n}{n!}\right)\nu_2 e^{-\nu_2 t}dt \\
&= \big((1-\xi)+\xi\widetilde{R}_1(s)\big)\widetilde{R}_2(s+\sigma_1(1-\widetilde{R}_1(s))) \\
&= \left((1-\xi)+\xi\frac{\nu_1}{\nu_1+s}\right)\frac{\nu_2}{\nu_2+s+\sigma_1\left(1-\frac{\nu_1}{\nu_1+s}\right)}.
\end{aligned}$$

The completion time C of a product, of which the distribution is represented by its Laplace-Stieltjes transform $\widetilde{C}(s)$, consists of an exponentially (μ_2) distributed service time B_2 with

the Laplace-Stieltjes transform $\widetilde{B}_2(s) = \frac{\mu_2}{\mu_2+s}$ and a Poisson $(\sigma_2 B_2)$ number of interruptions, each caused by a breakdown of M_2. Due to the interaction between the components, we know that both machines are operational at the start of the first completion time after a vacation period. As a result, the number of interfering M_1 repairs that occur during the first completion time is likely to be less than during later completion times. When ignoring this interaction effect and assuming that each breakdown has a duration that is distributed according to D and is independent of anything else, we obtain similarly to the computations above that

$$\widetilde{C}(s) \approx \widetilde{B}_2(s + \sigma_2(1 - \widetilde{D}(s))). \tag{6.20}$$

An application of the Fuhrmann-Cooper decomposition similar to the one encountered in Section 4.4 suggests that

$$g^{nprc} \approx \mathbb{E}[L_{\mathrm{M/G/1}}] + \mathbb{E}[L_{\mathrm{vac}}],$$

where $\mathbb{E}[L_{\mathrm{M/G/1}}] = \lambda_2\mathbb{E}[C] + \frac{\lambda_2\mathbb{E}[C^2]}{2(1-\lambda_2\mathbb{E}[C])}$ is the mean queue length of the number of products in an M/G/1 queue with Poisson (λ_2) arrivals and service times distributed according to the completion times C. Approximations for the moments of C follow by differentiation of (6.20) with respect to s. The term $\mathbb{E}[L_{\mathrm{vac}}]$ represents the expected queue length observed when the server is on a vacation, which is initiated any time the queue empties. This vacation period consists of periods of time where M_2 is operational, but may also consist of periods of time where M_2 is down in case a breakdown occurs before a type-2 product arrival. When conditioning on the event that M_2 is operational, we obviously have that the queue is empty. When conditioning on the event that M_2 is down, observe that under the assumption of independent and identically distributed downtimes, the expected queue length then amounts to the expected number of arrivals during the past part of a downtime D. The duration of this past part has expectation $\frac{\mathbb{E}[D^2]}{2\mathbb{E}[D]}$, where the moments of D can be computed by differentiation of $\widetilde{D}(s)$ with respect to s. Finally, we assert that the probability of the latter event occuring is closely approximated by $\frac{\mathbb{E}[D]}{\frac{1}{\sigma_2}+\mathbb{E}[D]}$, where $\frac{1}{\sigma_2}$ is the expected duration of an uptime of M_2. As a result,

$$\mathbb{E}[L_{\mathrm{vac}}] \approx \lambda_2 \frac{\mathbb{E}[D^2]}{2\mathbb{E}[D]} \frac{\mathbb{E}[D]}{\frac{1}{\sigma_2} + \mathbb{E}[D]} = \lambda_2 \frac{\sigma_2\mathbb{E}[D^2]}{2(1+\sigma_2\mathbb{E}[D])}.$$

By combining the results above, we obtain the approximation g_{app}^{nprc} as given in (6.19). Note that the application of the Fuhrmann-Cooper decomposition requires that the completion times are mutually independent. However, in our case, this requirement is not met, again due to the interaction between the components. For example, a very long completion time may imply that the last actual service period of M_2 has been longer than usual. In turn, this implies that M_2 has been in operation for some time. Thus, if a M_2-breakdown occurs in the next completion time, it is more likely than usual that M_1 is also down at that point. Due to this interdependence, the application of the Fuhrmann-Cooper decomposition also results in a computation error. However, all computation errors made share the same source, namely the interaction between the components and in particular the role of M_1. As we already saw in Conjecture 6.4.3, the influence of M_1 on the relative value function is likely to be limited, especially for states with a large number of products in the queue. Therefore, we expect this approximation to be accurate, especially for the purpose of deciding which of the two priority policies available performs best (see also Remark 6.6.1).

6.5 Derivation of a near-optimal policy

Based on the explicit expressions for the relative value functions of the static policy and the priority policy as obtained in the previous section, we derive a nearly optimal dynamic policy. We do so in Section 6.5.1 by applying the one-step policy improvement method on both the static policy and the priority policy. The resulting improved policies π^{oss}, π_1^{osp} and π_2^{osp} can then be used to construct a nearly optimal policy, as discussed in Section 6.5.2. By construction, this near-optimal policy is applicable in a broader range of parameter settings than each of the improved policies separately.

6.5.1 One-step policy improvement

One-step policy improvement is an approximation method that is distilled from the policy iteration algorithm in Markov decision theory. In policy iteration, one starts with an arbitrary policy π^{init} for which the relative value function V^{init} is known. Next, using these values, an improved policy π^{imp} can be obtained by performing a policy improvement step:

$$\pi^{imp}(\boldsymbol{s}) = \underset{\boldsymbol{a}\in\mathscr{A}_{\boldsymbol{s}}}{\arg\min} \left\{ \sum_{\boldsymbol{s}'\in\mathscr{S}} p_{\boldsymbol{a}}(\boldsymbol{s},\boldsymbol{s}') V^{init}(\boldsymbol{s}') \right\}, \tag{6.21}$$

i.e. the minimising action of $K^{init}(\boldsymbol{s})$ as defined in (6.3). If $\pi^{imp} = \pi^{init}$, the optimal policy has been found. Otherwise, the procedure can be repeated with the improved policy by setting $\pi^{init} := \pi^{imp}$, generating a sequence converging to the optimal policy. However, as the relative value function of the improved policy may not be known explicitly, subsequent iterations may have to be executed numerically. To avoid this problem, the one-step policy improvement method consists of executing the policy improvement step only once. In this case, the algorithm starts with a policy for which an expression for the relative value function is known. The resulting policy is then explicit and can act as a basis for approximation of the optimal policy. We now derive two one-step improved policies based on the results of the static policy and the priority policy as obtained in Section 6.4.

6.5.1.1 One-step policy improvement based on the static policy

In Section 6.4.1, we have found the relative value function V^{sta} for the class of static policies, in which each policy corresponds to a splitting parameter $p \in (0,1)$. As an initial policy for the one-step policy improvement, we take the policy which already performs best within this class with respect to the weighted number of products in the system. Thus, we take as an initial policy the static policy with splitting parameter

$$p^{oss} = \underset{p}{\arg\min}\{g_p^{sta} : p \in \mathscr{P}\}, \tag{6.22}$$

where g_p^{sta} is defined as in Theorem 6.4.2 and where $\mathscr{P} \subset (0,1)$ is the set of splitting parameters which satisfy the stability conditions in (6.5). Then, by performing one step

of policy improvement as given in (6.21), we obtain

$$\begin{aligned}
&\pi^{oss}(x_1,x_2,w_1,w_2)\\
&= \underset{(q_1,q_2)\in\mathscr{A}_{(x_1,x_2,w_1,w_2)}}{\arg\min} \{q_1\nu_1(V^{sta}_{p^{oss}}(x_1,x_2,1,w_2)-V^{sta}_{p^{oss}}(x_1,x_2,0,w_2))\\
&\qquad\qquad + q_2\nu_2(V^{sta}_{p^{oss}}(x_1,x_2,w_1,1)-V^{sta}_{p^{oss}}(x_1,x_2,w_1,0))\}. \qquad (6.23)
\end{aligned}$$

It is easily seen that $V^{sta}_{p^{oss}}(x_1,x_2,1,w_2)-V^{sta}_{p^{oss}}(x_1,x_2,0,w_2)$, as well as $V^{sta}_{p^{oss}}(x_1,x_2,w_1,1)-V^{sta}_{p^{oss}}(x_1,x_2,w_1,0)$, is non-positive for any state $(x_1,x_2,w_1,w_2)\in\mathscr{S}$ by observing that $\alpha_{2,i}\geq\alpha_{1,i}$ and $\alpha_{3,i}\geq 0$, $i=1,2$. This means that π^{oss} satisfies the properties mentioned in Section 6.3.1. Therefore, we can simplify (6.23) to

$$\begin{aligned}
&\pi^{oss}(x_1,x_2,w_1,w_2)\\
&= \underset{(q_1,q_2)\in\{(1-w_1,0),(0,1-w_2)\}}{\arg\min} \{q_1\nu_1(V^{sta}_{p^{oss}}(x_1,x_2,1,w_2)-V^{sta}_{p^{oss}}(x_1,x_2,0,w_2))\\
&\qquad\qquad + q_2\nu_2(V^{sta}_{p^{oss}}(x_1,x_2,w_1,1)-V^{sta}_{p^{oss}}(x_1,x_2,w_1,0))\}.
\end{aligned}$$

Substituting $V^{sta}_{p^{oss}}$ as obtained in Theorem 6.4.2 in this expression yields the following one-step improved policy:

$$\pi^{oss}(x_1,x_2,w_1,w_2)=\begin{cases}(0,0) & \text{if } w_1=w_2=1,\\ (1,0) & \text{if } w_1=1-w_2=0, \text{ or if } w_1w_2=0 \text{ and}\\ & c_1\nu_1((\alpha_{1,1}-\alpha_{2,1})x_1-\alpha_{3,1})\\ & \leq c_2\nu_2((\alpha_{1,2}-\alpha_{2,2})x_2-\alpha_{3,2}),\\ (0,1) & \text{otherwise}\end{cases} \qquad (6.24)$$

for $(x_1,x_2,w_1,w_2)\in\mathscr{S}$, where expressions for the α-coefficients are obtained by substituting the value for p in the expressions given in Theorem 6.4.2 by its optimised counterpart p^{oss}. Thus, whenever both machines are not operational, the one-step improved policy π^{oss} prescribes to repair the machine M_i for which $c_i\nu_i((\alpha_{1,i}-\alpha_{2,i})x_i-\alpha_{3,i})$ is smallest, when adhering to π^{oss}.

REMARK 6.5.1. If $\mathscr{P}$ is empty, there is no static policy available which results in a system with stable queues. In such circumstances, the static policy cannot be used as an initial policy for the one-step policy improvement approach. However, the priority policy as studied in Section 6.4.2 may still result in a stable system. If this is the case, the priority policy may act as an initial policy for the one-step policy improvement method. We study this alternative in the next section.

REMARK 6.5.2. Whenever $\mathscr{P}$ is not empty, the optimal splitting parameter p^{oss} is guaranteed to exist. As g^{sta}_p is a continuous function in p for $p\in\mathscr{P}$, the optimal splitting parameter p^{oss} is then a root of $\frac{d}{dp}g^{sta}_p$ in the domain $\mathscr{P}$. This derivative, which forms a sixth-order polynomial in p, defies the possibility of deriving an explicit expression for p^{oss}. For implementational purposes, however, this poses no significant problems, as such roots can be found numerically up to arbitrary precision with virtually no computation time needed.

6.5.1.2 One-step policy improvement based on the priority policy

Although an explicit expression for the relative value function V_i^{prio} is not available, we have identified enough of its characteristics in Section 6.4.2 to allow the use of a priority policy π_i^{prio} as an initial policy. We now show how to compute the one-step improved policy π_1^{osp} based on the priority policy π_1^{prio}, i.e. the priority policy where M_1 is the high-priority machine. Of course, the one-step improved policy π_2^{osp} based on the priority policy π_2^{prio} again follows by interchanging indices in the expressions below. The improvement step as given in (6.21) implies, after performing the same simplification as in the case of the static policy, that

$$\begin{aligned}&\pi_1^{osp}(x_1,x_2,w_1,w_2)\\ &= \underset{(q_1,q_2)\in\{(1-w_1,0),(0,1-w_2)\}}{\arg\min} \{q_1\nu_1(V_1^{prio}(x_1,x_2,1,w_2)-V_1^{prio}(x_1,x_2,0,w_2))\\ &\qquad\qquad + q_2\nu_2(V_1^{prio}(x_1,x_2,w_1,1)-V_1^{prio}(x_1,x_2,w_1,0))\}. \end{aligned} \tag{6.25}$$

The simplification is justified by the fact that $V_1^{prio}(x_1,x_2,1,w_2)-V_1^{prio}(x_1,x_2,0,w_2)$ and $V_1^{prio}(x_1,x_2,w_1,1)-V_1^{prio}(x_1,x_2,w_1,0)$ are obviously non-positive, since also under the priority policy it is always beneficial for the system to have a machine operational. Due to this, it is clear that $\pi_1^{osp}(x_1,x_2,w_1,w_2)$ in (6.25) resolves to $((1-w_1),(1-w_2))$ in case $w_1=w_2=1$, $w_1=1-w_2=1$ or $1-w_1=w_2=1$. However, for the case $w_1=w_2=0$, there are no expressions for $V_1^{prio}(x_1,x_2,1,0)-V_1^{prio}(x_1,x_2,0,0)$ and $V_1^{prio}(x_1,x_2,0,1)-V_1^{prio}(x_1,x_2,0,0)$ available. Due to their general intractability, we use the approximations for these differences as derived in (6.17) instead. By plugging these approximations into (6.25) in case $w_1=w_2=0$, we obtain with a slight abuse of notation that

$$\pi_1^{osp}(x_1,x_2,w_1,w_2)=\begin{cases}(1,0) & \text{if } w_1=1-w_2=0, \text{ or if } w_1w_2=0 \text{ and}\\ & \nu_1(c_1((\upsilon_1-\upsilon_2)x_1-\upsilon_3)-c_2\Delta_{1,0}x_2)\\ & \leq -c_2\Delta_{0,1}\nu_2x_2,\\ (0,1) & \text{otherwise,}\end{cases} \tag{6.26}$$

where the parameters υ_1, υ_2, υ_3, $\Delta_{1,0}$ and $\Delta_{0,1}$ are as defined in Section 6.4.2.1 and Conjecture 6.4.4, respectively.

REMARK 6.5.3. We have based π^{osp} on an approximation of the relative value function V^{prio} rather than an exact expression. Nevertheless, we have already argued in Section 6.4.2.2 that these approximations are accurate. Moreover, the argmin operator in (6.25) only checks which of the two arguments is smallest. Therefore, the improvement step is very robust against approximation errors, especially since both arguments share the same source of approximation error.

6.5.2 Resulting near-optimal policy

In the previous section, we have constructed the improved policies π^{oss}, π_1^{osp} and π_2^{osp} based on the static policy and the priority policy. However, the question remains which of

these policies should be followed by the repairman given a particular case of the model. In this section, we suggest a near-optimal policy, which chooses one of the three policies based on the model parameters. To this end, we now inspect these improved policies as well as their initial policies.

First, we observe that each of the improved policies satisfy the structural properties of the optimal policy. The improved policies π^{oss}, π_1^{osp} and π_2^{osp} each instruct the repairman to work at full capacity whenever at least one of the machines is down and therefore satisfy the non-idling property as derived in Section 6.3.1. Furthermore, when both of the machines are down, the improved policies base the action on threshold curves (or, in this case, threshold lines), so that they also satisfy the properties discussed in Section 6.3.2. As each of the improved policies satisfy the required properties, we base the decision on which of the improved policies to follow on their respective initial policies.

In terms of feasibility, the static policy and the priority policy complement each other. For any model instance, one can construct an improved static policy if there exists a static policy that results in stable queues; i.e. there exists a value $p \in (0,1)$ such that (6.5) holds. Similarly, an improved priority policy can be constructed if either (6.9) or (6.12) holds. There are cases of the model for which there is no stable static policy, whereas a stable priority policy exists. There are also cases for which the reverse holds true. In these cases, it is clear whether to use an improved static policy or an improved priority policy as a near-optimal policy. However, in case both of the approaches are feasible, other characteristics of the improved policies need to be taken into account.

In case the repairman would have no information to base his decision on (i.e. he has no knowledge about the state of the machines), it is easily seen that the optimal policy among the class of deterministic policies belongs to the class of static policies. The optimal policy in the current model, however, does not constitute a static policy, as the static policy does not have the non-idling property. This is the case because under the static policy, the server works at partial capacity when exactly one of the machines is down. Nevertheless, this problem does not arise with the improved version of the static policy.

As for the priority policy, if the load presented to the system would be such that the queues of products are never exhausted, it is easily seen that the optimal policy is in the class of priority policies. In such a case, the possibility of having a machine in an operational but idle state then disappears, so that the optimal policy always gives priority to one machine over the other due to faster service of products, a slower breakdown, faster repair times or a higher cost rate. We therefore expect the priority policy (and thus also its improved version) to work particularly well in our model when the model parameters are skewed in the favour of repair of a certain machine and when the queues of the products are particularly heavily loaded, such that the machines are almost never idling. The performance of the improved static policy, however, is not expected to be as sensitive to the load of the system, since the static policy balances the repair fractions based on, among other things, the load offered to each of the queues.

Based on the observations above, we suggest a near-optimal policy that is expressed in terms of a few simple decision rules. A schematic representation of this near-optimal policy is given in Figure 6.2. This near-optimal policy prescribes to follow the improved static policy as derived in Section 6.5.1.1 if there is a static policy available that results in a stable system (i.e. when there exists a p for which (6.5) holds). Otherwise, an improved priority policy should be followed, provided that a stable priority policy exists. In case only one of the priority policies is stable (i.e. either only (6.9) or only (6.12) holds), the

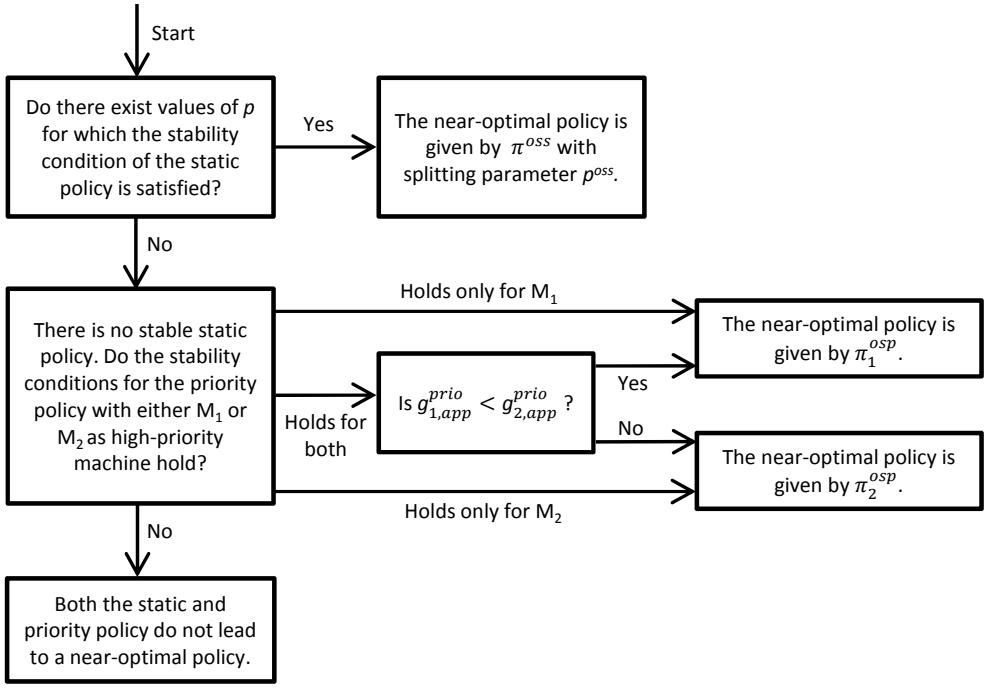

FIGURE 6.2: Schematic representation of the near-optimal policy.

choice of which improved priority policy to follow is easy. When both of them are stable, the choice is based on which of the two initial priority policies are expected to perform best. That is, the near-optimal policy will then select the improved policy corresponding to policy π_1^{prio}, if its approximated long-term average costs $g_{1,\,app}^{prio}$ as given in Approximation 6.4.5 is smaller than its equivalent $g_{2,\,app}^{prio}$ obtained by interchanging the indices in (6.18). Observe that this near-optimal policy is applicable in a wider range of parameter settings than each of the improved policies π^{oss}, π_1^{osp} and π_2^{osp} separately.

We end this section with several remarks concerning the obtained near-optimal policy.

REMARK 6.5.4. As the nearly optimal policy requires a stable static policy or a stable priority policy as a basis for one-step improvement, the approach only works when either (6.5) (for some value of $p \in (0,1)$), (6.9) or (6.12) holds. However, in theory, it is possible for some parameter settings that none of these conditions are satisfied, whereas stable policies do actually exist. However, one can reason that the parameter region where this occurs is fairly small. First, it is trivially seen that the stability condition (6.5) for the static policy only significantly differs from the necessary stability conditions given in (6.1) when the breakdown rates are large compared to the repair rates. In practice, however, breakdown rates are often much smaller than repair rates. Furthermore, for the priority policy, the conditions for λ_1 in (6.9) and λ_2 in (6.12) coincide with the requirements given in (6.1) for λ_1 and λ_2, respectively. Thus, the parameter region where our approach does not work only covers parameter settings where both λ_1 and λ_2 are close to their boundary values $\mu_1 \frac{\nu_1}{\sigma_1+\nu_1}$ and $\mu_2 \frac{\nu_2}{\sigma_2+\nu_2}$, respectively. Finally, we observe that (6.1) only presents

necessary conditions for the existence of a stable policy, but does not provide sufficient conditions. Therefore, the size of this parameter region is limited even further.

REMARK 6.5.5. Many optimisation approaches in Markov decision theory suffer from the curse of dimensionality. When dimensions are added to the state space, e.g. by adding more machines to the problem, the size of the state space increases considerably, so that numerical computation techniques break down due to time and resource constraints. Note, however, that the approach presented in this chapter generally scales well in the number of machines and the corresponding queues of products. The one-step improved policy based on the static policy can be modified to allow for models with $N > 2$ machines, since a decomposition of the system in the fashion of (6.6) can then be done into N components. After finding a vector of splitting parameters $(p_1^{oss}, p_2^{oss}, \ldots, p_N^{oss})$, the execution of the one-step policy improvement algorithm will then still result in a simple decision rule similar to (6.24). Likewise, the priority policy may be used to derive near-optimal policies in a model with larger dimensions. The current approximation for the relative value function V^{prio} in the case of $N = 2$ already accounts for the components containing the two most prioritised machines in a model with $N > 2$ machines, as the repair capacity assigned to a machine is not affected by the breakdown of a machine with lower priority. When approximations for the relative value function pertaining to lower prioritised components can be found, a nearly optimal policy follows similarly to the case $N = 2$.

6.6 Numerical study

In this section, we numerically assess the performance of the near-optimal policy obtained in Section 6.5 with respect to the optimal policy. We do this by comparing the average costs per time unit of both policies applied to a large number of model instances. To ensure that there is heavy competition between the machines for the resources of the repairman, we study instances with breakdown rates that are roughly of the same order as the repair rates. In these cases, the event that both machines are in need of repair is not a rare one, which allows us to compare the performance of the near-optimal policy to that of the optimal policy. We will see that the near-optimal policy performs very well over a wide range of parameter settings. Moreover, we observe several parameter effects. Throughout, we also give results for the improved static and priority policies (insofar as they exist) in order to observe how the near-optimal policy compares to these policies in terms of performance.

The complete test bed of instances that are analysed contains all 2916 possible combinations of the parameter values listed in Table 6.1. This table lists multiple values for the cost weights of having products in Q_1 and Q_2 (i.e. c_1 and c_2), the service rates at which M_1 and M_2 serve products when operational (i.e. μ_1 and μ_2), their breakdown rates (i.e. σ_1 and σ_2) as well as their repair rates (i.e. ν_1 and ν_2). Finally, the product arrival rates λ_1 and λ_2 are specified by the values of the parameters $\hat{\rho}_1^{FCFS}$ and $\hat{\rho}_2^{FCFS}$ given in the table, where $\hat{\rho}_i^{FCFS}$ represents the scaled load offered to M_i if the repairman would repair the machines in a first-come-first-served manner. More specifically, the arrival rates are taken such that the values of the scaled load

$$\hat{\rho}_i^{FCFS} = \frac{\lambda_i}{\mu_i m_{C,i}}$$

TABLE 6.1: Parameter values of the test bed.

Parameter	Considered parameter values
c_1	$\{0.25, 0.75\}$
c_2	$\{1\}$
$(\hat{\rho}_1^{FCFS}, \hat{\rho}_2^{FCFS})$	$a_i^\rho \cdot b_j^\rho \quad \forall i,j$, where $\boldsymbol{a}^\rho = \{0.25, 0.5, 0.75\}$ and $\boldsymbol{b}^\rho = \{(\frac{2}{3}, \frac{4}{3}), (1,1), (\frac{4}{3}, \frac{2}{3})\}$
(μ_1, μ_2)	$\{(0.75, 1.25), (1.25, 0.75), (1., 1.)\}$
(σ_1, σ_2)	$a_i^\sigma \cdot b_j^\sigma \quad \forall i,j$, where $\boldsymbol{a}^\sigma = \{0.1, 1\}$ and $\boldsymbol{b}^\sigma = \{(\frac{1}{2}, \frac{3}{2}), (1,1), (\frac{3}{2}, \frac{1}{2})\}$
(ν_1, ν_2)	$a_i^\nu \cdot b_j^\nu \quad \forall i,j$, where $\boldsymbol{a}^\nu = \{0.025, 0.1, 1\}$ and $\boldsymbol{b}^\nu = \{(\frac{1}{2}, \frac{3}{2}), (1,1), (\frac{3}{2}, \frac{1}{2})\}$

(cf. (2.2)) would coincide with those given in Table 6.1 if the repairman were to follow a first-come-first-served policy. Recall that $m_{C,i}$ represents the fraction of time that M_i is operational under a first-come-first-served policy. The values for $\hat{\rho}_i^{FCFS}$, σ_i and ν_i are varied in the order of magnitude through the values a_i^ρ, a_i^σ and a_i^ν as specified in the table and in the imbalance through the values b_j^ρ, b_j^σ and b_j^ν. For example, the load values $(\hat{\rho}_1^{FCFS}, \hat{\rho}_2^{FCFS})$ run from $(0.25 \cdot \frac{2}{3}, 0.25 \cdot \frac{4}{3}) = (\frac{1}{6}, \frac{1}{3})$, being small and putting the majority of the load on the second queue, to $(0.75 \cdot \frac{4}{3}, 0.75 \cdot \frac{2}{3}) = (1, 0.5)$, being large and putting the majority on the first queue. Observe that in the latter case, $\hat{\rho}_1^{FCFS}$ takes the value of one. Thus, we also consider cases where not all of the queues would be stable if the repairman would repair the machines in a first-come-first-served fashion.

For the systems corresponding to each of the parameter combinations in Table 6.1, it turns out that there is always at least one static policy or priority policy available as an initial policy, so that the near-optimal policy is feasible. We numerically compute the average costs $g^{n\text{-}opt}$ incurred per time unit by the system if the repairman were to follow the near-optimal policy as suggested in Section 6.5.2. Next to this, we also compute the average costs g^{opt} incurred per time unit if the repairman were to follow the optimal policy. We do this by using the value iteration algorithm (see e.g. [202]). Subsequently, we compute the relative difference $\Delta^{n\text{-}opt}$ between these approximations, i.e.

$$\Delta^{n\text{-}opt} = 100\% \times \frac{g^{n\text{-}opt} - g^{opt}}{g^{opt}}.$$

For instances where the corresponding initial policy exists, we also compute the relative differences of the improved policies considered in this chapter. That is, we compute similarly defined relative differences Δ^{oss} and Δ^{osp} for the improved static policy and the improved policy based on the priority policy with the smallest value for g_{app}^{prio} as computed in Approximation 6.4.5, respectively. Obviously, $\Delta^{n\text{-}opt}$, Δ^{oss} and Δ^{osp} cannot take negat-

TABLE 6.2: Percentual relative differences $\Delta^{n\text{-}opt}$, Δ^{oss} and Δ^{osp} categorised in bins.

	0-0.1%	0.1-1%	1-10%	10-25%	25%+
% of rel. differences $\Delta^{n\text{-}opt}$	36.73%	30.39%	32.55%	0.34%	0.00%
% of rel. differences Δ^{oss}	32.29%	32.44%	35.01%	0.26%	0.00%
% of rel. differences Δ^{osp}	57.95%	18.55%	16.43%	5.27%	1.80%

ive values. Furthermore, the closer these values are to zero, the better the corresponding policy performs.

In Table 6.2, the computed relative differences are summarised. We note that the vast majority of relative differences corresponding to the near-optimal policy do not exceed 10%, and more than half of the cases constitute a difference lower than 1%. These results show that the near-optimal policy works very well. The worst performance of the near-optimal policy encountered in the test bed is the exceptional case with parameters $(c_1, c_2) = (0.25,\ 1)$, $(\hat{\rho}_1^{FCFS}, \hat{\rho}_2^{FCFS}) = (1,\ 0.5)$, $(\mu_1, \mu_2) = (0.75,\ 1.25)$, $(\sigma_1, \sigma_2) = (0.15,\ 0.05)$ and $(\nu_1, \nu_2) = (0.05,\ 0.15)$. For this case, we found that $g^{opt} = 26.37$, $g^{n\text{-}opt} = 32.70$ and consequently $\Delta^{n\text{-}opt} = 24.05\%$. For this instance, any static policy, as well as the first-come-first-served policy, results in unstable queues. Moreover, this instance is characterised by highly asymmetric model parameters, but in such a way that neither of the machines would be a clear candidate for the role of the high-priority machine in the priority policy.

We also see in Table 6.2 that the improved static policy performs similarly to the near-optimal policy in terms of relative differences calculated. This is not surprising, as by construction, the near-optimal policy follows the improved static policy in case the latter exists. However, the gain of the near-optimal policy lies primarily in the fact that the near-optimal policy can handle a far broader range of parameter settings than the static policy. For example, of all instances with $\boldsymbol{a}^{\rho} = 0.75$, there are 268 instances for which the improved static policy is not available due to stability issues. The near-optimal policy, however, does result in an implementable policy for all 2916 instances considered in the test bed.

Judging by Table 6.2, the performance of the improved priority policy does differ from that of the near-optimal policy as opposed to the improved static policy. In 57.95% of the cases where an improved priority policy is available, the performance of the improved priority policy is less than 0.1% removed from that of the optimal policy. However, the relative difference exceeds 10% in more than 7% of the cases. Thus, there is far more variation in the performance of the priority policy than in the performance of the near-optimal policy. Furthermore, for 414 of the instances considered in this section, there is no improved priority policy available. Nevertheless, it is important to note that the set of instances for which no priority policy exists is completely disjoint of the set consisting of instances with no available improved static policy. This illustrates the fact that the improved static policy and the improved priority policy are complementary. These complementary parameter regions are combined in the near-optimal policy.

To observe any further parameter effects, Table 6.3 displays the mean relative difference $\Delta^{n\text{-}opt}$, Δ^{oss} and Δ^{osp} categorised in some of the variables. Based on these results, we identify four factors determining the quality of the near-optimal policy:

TABLE 6.3: Mean percentual relative differences categorised in each of the parameters as specified in Table 6.1.

(a)

c_1	0.25	0.75
$\Delta^{n\text{-}opt}$	1.26%	1.06%
Δ^{oss}	1.30%	1.14%
Δ^{osp}	2.23%	2.50%

(b)

a_i^{ν}	0.1	1
$\Delta^{n\text{-}opt}$	1.84%	0.48%
Δ^{oss}	1.89%	0.50%
Δ^{osp}	4.58%	1.06%

(c)

a_i^{ρ}	0.25	0.5	0.75
$\Delta^{n\text{-}opt}$	0.66%	1.00%	1.83%
Δ^{oss}	0.66%	1.01%	2.22%
Δ^{osp}	2.63%	1.30%	3.24%

(d)

a_i^{σ}	0.025	0.1	1
$\Delta^{n\text{-}opt}$	0.42%	0.77%	2.29%
Δ^{oss}	0.48%	0.76%	2.32%
Δ^{osp}	0.07%	2.21%	6.58%

(e)

(μ_1, μ_2)	(0.75, 1.25)	(1, 1)	(1.25, 0.75)
$\Delta^{n\text{-}opt}$	1.26%	1.15%	1.07%
Δ^{oss}	1.31%	1.22%	1.14%
Δ^{osp}	2.38%	2.38%	2.40%

(f)

b_j^{ρ}	$(\frac{2}{3}, \frac{4}{3})$	(1, 1)	$(\frac{4}{3}, \frac{2}{3})$
$\Delta^{n\text{-}opt}$	1.39%	0.98%	1.11%
Δ^{oss}	1.58%	0.99%	1.13%
Δ^{osp}	1.51%	3.99%	1.77%

(g)

b_j^{σ}	$(\frac{1}{2}, \frac{3}{2})$	(1, 1)	$(\frac{3}{2}, \frac{1}{2})$
$\Delta^{n\text{-}opt}$	1.09%	1.07%	1.31%
Δ^{oss}	1.19%	1.13%	1.35%
Δ^{osp}	2.12%	1.90%	3.12%

(h)

b_j^{ν}	$(\frac{1}{2}, \frac{3}{2})$	(1, 1)	$(\frac{3}{2}, \frac{1}{2})$
$\Delta^{n\text{-}opt}$	1.39%	1.20%	0.90%
Δ^{oss}	1.39%	1.28%	0.99%
Δ^{osp}	3.24%	1.70%	2.23%

- Table 6.3(a) suggests that the closer the value of c_1 is to the value of c_2, the better the performance of the near-optimal policy becomes. A similar effect can be observed in Table 6.3(f) with the values $\hat{\rho}_1^{FCFS}$ and $\hat{\rho}_2^{FCFS}$. These effects suggest that the level of asymmetry in the parameters plays a role in the effectiveness of the near-optimal policy. Intuitively, this makes sense, as the optimal policy gets easier to predict when the system becomes more symmetric. For example, in the case of a completely symmetric model (i.e. $\lambda_1 = \lambda_2$, $\mu_1 = \mu_2$ etc.), the threshold curve of the optimal policy is easily seen to be the line $x_1 = x_2$ by a switching argument. In that case, the improved static policy also attains this curve, which suggests that the near-optimal policy is optimal in symmetric systems, provided that the initial static policy is stable.

- Judging by Table 6.3(c), the performance of the near-optimal policy with respect to the optimal policy becomes worse when the load of products offered to the queues increases. This can be explained by the fact that in case of a smaller load, products on average encounter less waiting products in their respective queue and are therefore less influenced by the downtimes of their machines which occurred before their arrival. In turn, this means that the sojourn time of products in the system is less sensitive to any suboptimal decisions taken in the past, improving the accuracy of the near-optimal

policy. In the extreme case where the load offered to each queue equals zero (i.e. there are no products arriving), any policy is optimal, as the system does not incur any costs in that case.

- From Tables 6.3(b) and 6.3(d), it is apparent that the quality of the near-optimal policy is influenced by the values of a_i^ν and a_i^σ. This can be explained mainly by the fact that these values determine the level of competition between the machines for access to the repairman. When breakdowns do not occur often and repairs are done quickly, the event of having both machines down is exceptional, so that any suboptimality of the policy used is expected to have a relatively little impact on the average costs.
- Tables 6.3(e) and 6.3(h) seem to contradict the first observation that a high level of symmetry in the system improves the performance of the near-optimal policy, as the near-optimal policy now seems to perform better when M_2 has more 'favourable' characteristics with respect to M_1. In other words, the fast product services and fast repairs of M_2 make it lucrative to repair M_2 at the expense of additional downtime for M_1. However, note that this effect occurs because the cost weights are already taken in favour of the repair of M_2 in every instance of the test bed. When the loads are such that the static policy becomes infeasible, a priority policy with M_2 as the high-priority machine will then already be close to optimal. Therefore, its improved version also works particularly well. However, if, as opposed to the cost weights, the rates of product services, breakdowns and repairs are in favour of M_1, a priority policy works less well, since there is no clear candidate for the high-priority machine any more. This leaves room for suboptimality of the improved priority policy.

As for the other policies, Table 6.3 suggests that the performance of the improved static policy exhibits similar parameter effects to that of the near-optimal policy. Again, this is not surprising considering the way the near-optimal policy is constructed. However, the improved priority policy behaves differently in a number of ways. First, Tables 6.3(a) and 6.3(f) show that the improved priority policy performs better in systems with skewed model parameters. For these systems, the operational state of one machine is generally evidently more important than the other, so that the initial priority policy already performs quite well. Unlike the near-optimal policy and the improved static policy, we see in Table 6.3(c) that the performance of the improved priority policy does not necessarily increase in the load offered to the system. Finally, Table 6.3(e) suggests that the performance of the improved priority policy is highly insensitive to any difference in the service rates of the machines.

REMARK 6.6.1. In Section 6.4.2.2, we introduced an approximation g_{app}^{prio} for the long-term average costs of the priority policy with either machine as the high-priority machine. We did this for the purpose of predicting which of the two improved priority policies performs best in case both of them exist. Of the 2916 instances considered in the test bed, there are 1782 instances for which both priority policies lead to an improved policy. For each of these instances, it turns out that the best-performing improved priority policy corresponds to the initial priority policy with the smallest approximated costs. This suggests that the approximation for the long-term average costs fulfills its purpose well.

REMARK 6.6.2. In this section, we have considered models which consist of two machines and have breakdown rates and repair rates that are of a comparable size. Interference between machines, however, may in practice also occur in systems with a large number

of machines that have breakdown rates which are much smaller than their repair rates. In that case, having two or more machines in need of repair is not a rare event, so that the question of how to allocate the repairman's resources is still an important one. For models with a larger number of machines (and thus a larger number of queues), we have already established in Remark 6.5.5 that the near-optimal policy scales well. However, numerical computation techniques break down, so that a numerical study similar to the one in this section for $N = 2$ becomes infeasible. Nevertheless, observe that if N increases, but the breakdown rates decrease at a similar intensity, the average number of machines that are in need of repair fluctuates around the same mean. The situation of $N = 2$ and similar rates as considered in this section is thus comparable to the case of a large number of machines with dissimilar rates. Therefore, we expect that the near-optimal policy also performs well for the case of $N > 2$.

Appendix

6.A Proof of Proposition 6.3.1

PROOF. The proof is based on induction and the guaranteed convergence of the value iteration algorithm. We initially pick the function $V_0(s) = 0$ for all $s \in \mathscr{S}$. Obviously, this function satisfies properties 1 and 2. We show that these properties are preserved when performing one step of the value iteration algorithm. In mathematical terms, we show for any $n \in \mathbb{N}$ that the function V^{n+1} defined by $V^{n+1}(s) = H^n(s) + K^n(s)$ also satisfies the properties if V^n does. Because of the guaranteed convergence, V^{opt} then satisfies properties 1 and 2 by induction. For an extensive discussion of this techique to prove structural properties of relative value functions, see [147].

The induction step is performed as follows. We assume that properties 1 and 2 hold for V^n (the induction assumption). We will show that properties 1 and 2 hold for V^{n+1}. For the first property, observe that by interchanging the indices of the model parameters, one obtains another instance of the same model, since the structure of the model is symmetric. Therefore, the left-hand side of property 1 implies the right-hand side. To prove the left-hand side of property 1, we expand $V^{n+1}(x_1, x_2, 0, w_2) - V^{n+1}(x_1, x_2, 1, w_2)$ into V^n:

$$\begin{aligned} &V^{n+1}(x_1, x_2, 0, w_2) - V^{n+1}(x_1, x_2, 1, w_2) \\ &= H^n(x_1, x_2, 0, w_2) - H^n(x_1, x_2, 1, w_2) + K^n(x_1, x_2, 0, w_2) - K^n(x_1, x_2, 1, w_2). \end{aligned} \tag{6.27}$$

By rearranging the terms arising from (6.2) and applying the induction assumption, we have that

$$\begin{aligned} &H^n(x_1, x_2, 0, w_2) - H^n(x_1, x_2, 1, w_2) \\ &= \lambda_1(V^n(x_1 + 1, x_2, 0, w_2) - V^n(x_1 + 1, x_2, 1, w_2)) \\ &\quad + \lambda_2(V^n(x_1, x_2 + 1, 0, w_2) - V^n(x_1, x_2 + 1, 1, w_2)) \\ &\quad + \mu_1(V^n(x_1, x_2, 1, w_2) - V^n((x_1 - 1)^+, x_2, 1, w_2)) \\ &\quad + \mu_2 w_2(V^n(x_1, (x_2 - 1)^+, 0, w_2) - V^n(x_1, (x_2 - 1)^+, 1, w_2)) \\ &\quad + \sigma_2 w_2(V^n(x_1, x_2, 0, 1) - V^n(x_1, x_2, 1, 1)) \\ &\quad + (1 - \lambda_1 - \lambda_2 - \sigma_1 - w_2(\mu_2 + \sigma_2))(V^n(x_1, x_2, 0, w_2) - V^n(x_1, x_2, 1, w_2)) \end{aligned}$$

$$\geq (1-\lambda_1-\lambda_2-\sigma_1-w_2(\mu_2+\sigma_2))(V^n(x_1,x_2,0,w_2)-V^n(x_1,x_2,1,w_2)). \quad (6.28)$$

Furthermore, since the difference $V^n(x_1,x_2,w_1,1)-V^n(x_1,x_2,w_1,0)$, as well as the difference $V^n(x_1,x_2,1,w_2)-V^n(x_1,x_2,0,w_2)$, evaluates to a non-positive number, we can limit the set of possible minimising actions in K^n (see (6.3)) to $\{(q_1,q_2): q_1 \in \{0,1-w_1\} \wedge q_2 \in \{0,1-w_2\} \wedge q_1+q_2 = 1-w_1w_2\}$. By this and (6.3), we obtain

$$\begin{aligned}
&K^n(x_1,x_2,0,w_2)-K^n(x_1,x_2,1,w_2)\\
&=\min\{\nu_1(V^n(x_1,x_2,1,w_2)-V^n(x_1,x_2,0,w_2)),\\
&\qquad (1-w_2)\nu_2(V^n(x_1,x_2,0,1)-V^n(x_1,x_2,0,0))\}\\
&\quad -(1-w_2)\nu_2(V^n(x_1,x_2,1,1)-V^n(x_1,x_2,1,0)), \qquad (6.29)
\end{aligned}$$

where the second equality holds because of the induction assumption. Let E_1 denote the event that the last minimum is only minimised by its first argument and let E_2 be its complementary event. As a conclusion we find by combining (6.27)-(6.29) that

$$\begin{aligned}
&V^{n+1}(x_1,x_2,0,w_2)-V^{n+1}(x_1,x_2,1,w_2)\\
&\geq (1-\lambda_1-\lambda_2-w_2(\mu_2+\sigma_2)-\mathbb{1}_{\{E_1\}}\nu_1-\mathbb{1}_{\{E_2\}}(1-w_2)\nu_2)\\
&\qquad \times (V^n(x_1,x_2,0,w_2)-V^n(x_1,x_2,1,w_2))\\
&\quad +\mathbb{1}_{\{E_1\}}(1-w_2)\nu_2(V^n(x_1,x_2,1,0)-V^n(x_1,x_2,1,1))\\
&\quad +\mathbb{1}_{\{E_2\}}(1-w_2)\nu_2(V^n(x_1,x_2,0,1)-V^n(x_1,x_2,1,1))\\
&\geq 0.
\end{aligned}$$

The last inequality holds by applying the induction assumption on each term of the expression in front of it and observing, for the first term, that $(1-\lambda_1-\lambda_2-w_2(\mu_2+\sigma_2)-\mathbb{1}_{\{E_1\}}\nu_1-\mathbb{1}_{\{E_2\}}(1-w_2)\nu_2)$ is non-negative due to the uniformisation. This proves property 1.

We now turn to property 2. Note that also for property 2, the left-hand side implies the right-hand side due to symmetry arguments. To prove the left-hand side of property 2, we expand $V^{n+1}(x_1+1,x_2,w_1,w_2)-V^{n+1}(x_1,x_2,w_1,w_2)$ into V^n:

$$\begin{aligned}
&V^{n+1}(x_1+1,x_2,w_1,w_2)-V^{n+1}(x_1,x_2,w_1,w_2)\\
&=H^n(x_1+1,x_2,w_1,w_2)-H^n(x_1,x_2,w_1,w_2)+K^n(x_1+1,x_2,0,w_2)\\
&\quad -K^n(x_1,x_2,w_1,w_2). \qquad (6.30)
\end{aligned}$$

A lower bound for the H terms can be found by rearranging terms stemming from (6.2):

$$\begin{aligned}
&H^n(x_1+1,x_2,w_1,w_2)-H^n(x_1,x_2,w_1,w_2)\\
&=c_1+\lambda_1(V^n(x_1+2,x_2,w_1,w_2)-V^n(x_1,x_2,w_1,w_2))\\
&\quad +\lambda_2(V^n(x_1+1,x_2+1,w_1,w_2)-V^n(x_1,x_2,w_1,w_2))\\
&\quad +\mu_1w_1(V^n(x_1,x_2,w_1,w_2)-V^n((x_1-1)^+,x_2,w_1,w_2))\\
&\quad +\mu_2w_2(V^n(x_1+1,(x_2-1)^+,w_1,w_2)-V^n(x_1,(x_2-1)^+,w_1,w_2))\\
&\quad +\sigma_1w_1(V^n(x_1+1,x_2,0,w_2)-V^n(x_1,x_2,0,w_2))\\
&\quad +\sigma_2w_2(V^n(x_1+1,x_2,w_1,0)-V^n(x_1,x_2,w_1,0))\\
&\quad +(1-\lambda_1-\lambda_2-w_1(\mu_1+\sigma_1)-w_2(\mu_2+\sigma_2))
\end{aligned}$$

$$\begin{aligned}
&\times (V^n(x_1+1,x_2,w_1,w_2)-V^n(x_1,x_2,w_1,w_2))\\
\geq\ &(1-\lambda_1-\lambda_2-w_1(\mu_1+\sigma_1)-w_2(\mu_2+\sigma_2))\\
&\times (V^n(x_1+1,x_2,w_1,w_2)-V^n(x_1,x_2,w_1,w_2)),
\end{aligned} \tag{6.31}$$

where the inequality is easily seen to hold true due to the induction assumption. For the K terms, we can again limit the set of possible minimising actions to $\{(q_1,q_2) : q_1 \in \{0,1-w_1\}, q_2 \in \{0,1-w_2\}, q_1+q_2 = 1-w_1w_2\}$. By (6.3), we then have

$$\begin{aligned}
&K^{n+1}(x_1+1,x_2,w_1,w_2)-K^{n+1}(x_1,x_2,w_1,w_2)\\
&=\min\{(1-w_1)\nu_1(V^n(x_1+1,x_2,1,w_2)-V^n(x_1+1,x_2,0,w_2)),\\
&\qquad (1-w_2)\nu_2(V^n(x_1+1,x_2,w_1,1)-V^n(x_1+1,x_2,w_2,0))\}\\
&\quad -\min\{(1-w_1)\nu_1(V^n(x_1,x_2,1,w_2)-V^n(x_1,x_2,0,w_2)),\\
&\qquad (1-w_2)\nu_2(V^n(x_1,x_2,w_1,1)-V^n(x_1,x_2,w_1,0))\}.
\end{aligned} \tag{6.32}$$

We now show that $V^{n+1}(x_1+1,x_2,w_1,w_2)-V^{n+1}(x_1,x_2,w_1,w_2) \geq 0$ by combining (6.30)-(6.32) for every possible combination of w_1 and w_2 separately.

- For $w_1 = w_2 = 0$, we have

$$\begin{aligned}
&V^{n+1}(x_1+1,x_2,0,0)-V^{n+1}(x_1,x_2,0,0)\\
&\geq (1-\lambda_1-\lambda_2)(V^n(x_1+1,x_2,w_1,w_2)-V^n(x_1,x_2,w_1,w_2))\\
&\quad +\min\{\nu_1(V^n(x_1+1,x_2,1,0)-V^n(x_1+1,x_2,0,0)),\\
&\qquad \nu_2(V^n(x_1+1,x_2,0,1)-V^n(x_1+1,x_2,0,0))\}\\
&\quad -\min\{\nu_1(V^n(x_1,x_2,1,0)-V^n(x_1,x_2,0,0)),\\
&\qquad \nu_2(V^n(x_1,y_1,0,1)-V^n(x_1,y_1,0,0))\}.
\end{aligned}$$

  Due to the induction assumption, the arguments of both minimum operators are all negative. If it would be optimal to repair M_1 in the state $(x_1+1,x_2,0,0)$, the first argument of the first minimum is the minimising argument. The expression above then reduces to

$$\begin{aligned}
&V^{n+1}(x_1+1,x_2,0,0)-V^{n+1}(x_1,x_2,0,0)\\
&\geq (1-\lambda_1-\lambda_2)(V^n(x_1+1,x_2,0,0)-V^n(x_1,x_2,0,0))\\
&\quad +\nu_1(V^n(x_1+1,x_2,1,0)-V^n(x_1+1,x_2,0,0))\\
&\quad -\min\{\nu_1(V^n(x_1,x_2,1,0)-V^n(x_1,x_2,0,0)),\\
&\qquad \nu_2(V^n(x_1,y_1,0,1)-V^n(x_1,y_1,0,0))\}\\
&\geq (1-\lambda_1-\lambda_2)(V^n(x_1+1,x_2,0,0)-V^n(x_1,x_2,0,0))\\
&\quad +\nu_1(V^n(x_1+1,x_2,1,0)-V^n(x_1+1,x_2,0,0))\\
&\quad -\nu_1(V^n(x_1,x_2,1,0)-V^n(x_1,x_2,0,0))\\
&=(1-\lambda_1-\lambda_2-\nu_1)(V^n(x_1+1,x_2,0,0)-V^n(x_1,x_2,0,0))\\
&\quad +\nu_1(V^n(x_1+1,x_2,1,0)-V^n(x_1,x_2,1,0))\\
&\geq 0,
\end{aligned}$$

  where the last inequality follows from the induction assumption. In a similar way, it can be shown that $V^{n+1}(x_1+1,x_2,0,0)-V^{n+1}(x_1,x_2,0,0) \geq 0$ if it would be optimal to repair M_2 in the state $(x_1,x_2+1,0,0)$, exhausting all possible actions.

- If $w_1 = 0$ and $w_2 = 1$, we have

$$\begin{aligned}
&V^{n+1}(x_1+1,x_2,0,1) - V^{n+1}(x_1,x_2,0,1)\\
&\geq (1-\lambda_1-\lambda_2-\mu_2-\sigma_2)(V^n(x_1+1,x_2,0,1) - V^n(x_1,x_2,0,1))\\
&\quad + \nu_1(V^n(x_1+1,x_2,1,1) - V^n(x_1+1,x_2,0,1))\\
&\quad - \nu_1(V^n(x_1,x_2,1,1) - V^n(x_1,x_2,0,1))\\
&= (1-\lambda_1-\lambda_2-\mu_2-\sigma_2-\nu_1)(V^n(x_1+1,x_2,0,1) - V^n(x_1,x_2,0,1))\\
&\quad + \nu_1(V^n(x_1+1,x_2,1,1) - V^n(x_1,x_2,1,1))\\
&\geq 0,
\end{aligned}$$

  where the last inequality follows from the induction assumption.

- The case $w_1 = 1 - w_2 = 1$ is handled similarly to the case $w_1 = 1 - w_2 = 0$.
- When $w_1 = w_2 = 1$, we have

$$\begin{aligned}
&V^{n+1}(x_1+1,x_2,1,1) - V^{n+1}(x_1,x_2,1,1)\\
&\geq \left(1 - \sum_{i=1}^{2}(\lambda_i+\mu_i+\sigma_i)\right)(V^n(x_1+1,x_2,1,1) - V^n(x_1,x_2,1,1)),
\end{aligned}$$

  which is easily seen to be non-negative by the induction assumption.

Putting together all four combinations, we have proved that V^{n+1} satisfies property 2. As V^{n+1} satisfies properties 1 and 2, V^{opt} does too by an induction argument. □

Part II

The Markovian polling model

7

TWO-QUEUE EXHAUSTIVE MODELS

In this chapter, we start the analysis of the Markovian polling model as described in Section 1.3.2 by studying the two-queue subclass. Furthermore, we assume that the server only initiates a switch-over period to another queue as soon as the queue he is currently visiting is completely empty. Under these assumptions, we derive an expression for the probability generating function of the joint queue length distribution at polling epochs (i.e. the beginnings of a visit period). Based on these results, we obtain explicit expressions for the Laplace-Stieltjes transforms of the *complete* waiting-time distributions and the probability generating function of the *complete* joint queue length distribution at an arbitrary point in time. We also study the heavy-traffic behaviour of these distributions, which results in compact and closed-form expressions for the distribution functions themselves. The heavy-traffic behaviour turns out to be similar to that of cyclic polling models, provides insights into the main effects of the model parameters when the system is heavily loaded and can be used to derive closed-form approximations for the waiting-time distribution or the queue length distribution.

7.1 Introduction

In this chapter, we consider the special setting of two-queue Markovian polling models, where the queues are served exhaustively (i.e. the server will only start a switch-over period if the current queue is completely empty). As we have observed in Section 1.3.2, Markovian polling models are hard to analyse, since they typically do not satisfy the so-called *branching property*. In other words, for general Markovian polling models, the queue length vectors at successive times when the server starts a visit period do not form a multi-type branching process with immigration. However, it turns out that this property does hold for the special setting of exhaustive two-queue models, as will be described in greater detail in Remark 7.4.3. This allows for the derivation of explicit expressions for (transforms of) the *complete* waiting-time and queue length distributions.

Initially, we will be concerned with the waiting-time and queue length distributions when the load offered to the server is such that the queues are stable. The analysis of non-trivial two-queue polling systems, such as [50], oftentimes includes a solution to a Riemann-Hilbert boundary value problem. We, however, follow an approach similar to the analysis of [272], which uses a recursive iteration of a functional equation for the

probability generating function of the joint queue length distribution at moments the server starts a visit period, and therefore avoids such a boundary value problem.

We also study the behaviour of the system in a heavy-traffic regime, i.e. when the load offered to the server is scaled to such a proportion that the queues are on the verge of instability. Many techniques have been proposed to obtain the heavy-traffic behaviour of polling models. Initial studies for cyclic polling models can be found in [65, 66], where the occurrence of a so-called heavy-traffic averaging principle is established. This principle implies that, although the total scaled load in the system tends to a Bessel-type diffusion in the heavy-traffic regime, the total load in the system may be considered as a constant during the course of a polling cycle, while the loads of the individual queues fluctuate like in a fluid model. In [248], several heavy-traffic limits have been established for models with a first-come-first-served scheduling discipline by taking limits in known expressions for the Laplace-Stieltjes transform of the waiting-time distribution. This method has also been used in [P1, P2] to derive heavy-traffic results for models with scheduling disciplines other than the first-come-first-served discipline. Alternatively, for the first-come-first-served case, [184] derives the heavy-traffic results obtained in [248] in a somewhat more general setting by studying the behaviour of the descendant set approach (a numerical computation method, cf. [145]) in the heavy-traffic limit. Another tool in the heavy-traffic analysis of polling models is branching theory, theorems of which led to heavy-traffic results in [249]. Other methods for obtaining heavy-traffic behaviour include perturbation techniques, which have been exploited in [44] to study a specific class of non-branching polling models, and mean-value analysis (cf. [252]). In our heavy-traffic analysis, we partly use the key ideas of [184].

The remainder of this chapter is structured as follows. In Section 7.2, we introduce the two-queue Markovian polling model more carefully, and we provide the necessary notation. Then, under the assumption of a stable system, we obtain explicit expressions for several performance measures of the two-queue Markovian polling model with exhaustive service in Section 7.3. In particular, we derive explicit expressions for (transforms of) the waiting-time distributions and the joint queue length distribution by taking a functional equation for the probability generating function of the joint queue length distribution at polling epochs as a starting point. Although these expressions consist of infinite products and are thus not in closed form, the products converge fast so that truncation leads to accurate approximations. We also consider the behaviour of the waiting-time and queue length distributions in a heavy-traffic regime in Section 7.4. From a theoretical perspective, these results are interesting, since, unlike previous studies, the complete distributions of the waiting times and queue lengths are analysed. The results in this chapter are only proved for the two-queue exhaustive case, and are not easily extendable to more general assumptions. Nevertheless, they may offer some insights into the general case. For instance, we will show that, except for some minor adjustments, the heavy-traffic behaviour of two-queue Markovian polling models with exhaustive service is similar to that of cyclic polling models as derived in the literature. It seems that this relation also exists under more general assumptions, as we will conclude in Remark 7.4.4. From a practical perspective, the results are useful, as they not only provide closed-form approximations for several performance measures that perform well when the system is heavily loaded (as is usual in practice), but also give insights into the key effects of the model parameters on the waiting times and queue lengths.

7.2 Model description and notation

We study a special case of the model as described in Section 1.3.2, which consists of two infinite-buffer queues, Q_1 and Q_2, and a single server. Customers arriving at Q_i, also referred to as type-i customers, do so according to a Poisson process with intensity λ_i. The generic service requirement of a type-i customer is represented by the random variable B_i, of which the Laplace-Stieltjes transform is given by $\widetilde{B}_i(s) = \mathbb{E}[e^{-sB_i}]$, and the moments $\mathbb{E}[B_i^k]$, $k \geq 1$, are assumed to be finite. The load that Q_i brings to the system is denoted by $\rho_i = \lambda_i \mathbb{E}[B_i]$. The aggregate load offered to the server is denoted by $\rho = \rho_1 + \rho_2$. Initially, we study the case where the aggregate load is less than one, so that the queues are stable. After that, we study the system in a so-called *heavy-traffic* regime: the case where ρ tends to one, i.e. the point at which the queues are at the verge of instability.

The single server can only serve one queue at a time. Hence, after serving a given number of customers at one queue in the order of arrival (a visit period), the server commences a switch-over period to initiate a new visit period at any queue. Such a setup takes a random amount of time. In most studies on two-queue polling systems, it is assumed that the server visits the queues in an alternating order. We, however, adopt a more general server routing mechanism. We assume that when the server completes a visit period at Q_1, he commences with probability $\xi_1 \in [0,1)$ a switch-over period to set up for yet another visit period at Q_1. In the other case (which occurs with probability $1-\xi_1$), the server sets up for a visit to Q_2. Similarly, after visiting Q_2, the server prepares for another visit period at Q_2 with probability $\xi_2 \in [0,1)$. Otherwise, he will set up for service at Q_1. This particular routing regime covers the alternating routing regime by taking $\xi_1 = \xi_2 = 0$.

Observe that this routing mechanism falls in the class of Markovian routing mechanisms, since the position of the server is governed by a two-state discrete-time Markov chain of which the transition matrix has diagonal elements ξ_1 and ξ_2. By calculating the limiting distribution of this Markov chain, one finds that a fraction $q_1 = \frac{1-\xi_2}{2-\xi_1-\xi_2}$ of the switch-over periods correspond to setups to Q_1 and the remaining fraction $q_2 = \frac{1-\xi_1}{2-\xi_1-\xi_2}$ are setups to Q_2. The probability $v_{i,j}$ that, provided the server is currently visiting Q_j, the server visited Q_i during the previous visit period follows straightforwardly from these computations. It is trivial to see that $v_{1,1} + v_{2,1} = 1$ and $v_{1,2} + v_{2,2} = 1$. In particular, we have that

$$v_{1,1} = \frac{\xi_1 q_1}{\xi_1 q_1 + (1-\xi_2) q_2} = \xi_1, \quad v_{1,2} = \frac{(1-\xi_1) q_1}{(1-\xi_1) q_1 + \xi_2 q_2} = 1 - \xi_2,$$
$$v_{2,1} = \frac{(1-\xi_2) q_2}{\xi_1 q_1 + (1-\xi_2) q_2} = 1 - \xi_1 \text{ and } v_{2,2} = \frac{(1-\xi_1) q_1}{(1-\xi_1) q_1 + \xi_2 q_2} = \xi_2.$$

Over the course of a visit period, the server serves the queues in an exhaustive manner. In other words, the server will completely empty a queue during a visit period, before he commences a switch-over period. To gain more insight in the dynamics of the exhaustive service discipline, let Γ_i denote the duration of a busy period in an M/G/1 queue with the same arrival process and service time distribution as Q_i. This busy period consists of the service of its first customer, the services of the customers arriving during the service of the first customer (i.e. the 'children'), the services of the customers arriving during the service of the children (i.e. the 'grandchildren') and so forth. The Laplace-Stieltjes

transform corresponding to Γ_i, denoted by $\widetilde{\Gamma}_i(s) = \mathbb{E}[e^{-s\Gamma_i}]$, is well known to satisfy the functional equation

$$\widetilde{\Gamma}_i(s) = \widetilde{B}_i(s + \lambda_i(1 - \widetilde{\Gamma}_i(s))). \tag{7.1}$$

We denote the number of customers that arrive at Q_j over the course of a busy period at Q_i with $K_{i,j}$, $i \neq j$. Its probability generating function $\widetilde{K}_{i,j}(z) = \mathbb{E}[z^{K_{i,j}}]$ is given by

$$\widetilde{K}_{i,j}(z) = \sum_{k=0}^{\infty} z^k \int_{t=0}^{\infty} e^{-\lambda_j t} \frac{(\lambda_j t)^k}{k!} d\mathbb{P}(\Gamma_i < t) = \widetilde{\Gamma}_i(\lambda_j(1-z)).$$

If a server starts a visit period at Q_i when there are n customers in that queue, the duration of that visit period is the n-fold convolution of Γ_i. It is important to note that if the server sets up for service at the same queue afterwards, Q_i is not necessarily empty at the start of the new visit period, as customers may have arrived over the course of the intermediate switch-over period.

We assume the distribution of the durations of the switch-over periods to depend on the queue the server just visited as well as the destination queue. In particular, we assume that a setup from Q_i to Q_j takes a continuously distributed stochastic amount of time $S_{i,j}$, of which the Laplace-Stieltjes transform is given by $\widetilde{S}_{i,j}(s) = \mathbb{E}[e^{-sS_{i,j}}]$, $i, j \in \{1, 2\}$. The average duration of an arbitrary switch-over period incurred by the server is given by $\sigma = \sum_{i=1}^{2} \sum_{j=1}^{2} \nu_{i,j} q_j \mathbb{E}[S_{i,j}]$. Let $M_{i,j}^{(k)}$ be the number of arriving type-k customers over the course of a switch-over period from Q_i to Q_j. Similar to the computation of $\widetilde{K}_{i,j}(z)$, it can then be derived that the two-dimensional probability generating function $\widetilde{M}_{i,j}(z_1, z_2) = \mathbb{E}[\prod_{k=1}^{2} z_k^{M_{i,j}^{(k)}}]$ is given by

$$\begin{aligned}\widetilde{M}_{i,j}(z_1, z_2) &= \int_{t=0}^{\infty} \sum_{n_1=0}^{\infty} \sum_{n_2=0}^{\infty} \prod_{k=1}^{2} \left(z_k^{n_k} e^{-\lambda_k t} \frac{(\lambda_k t)^{n_k}}{n_k!} \right) d\mathbb{P}(S_{i,j} < t) \\ &= \widetilde{S}_{i,j}(\lambda_1(1 - z_1) + \lambda_2(1 - z_2)).\end{aligned}$$

We assume all interarrival times, service times and switch-over times to be independent.

In the remainder of this chapter, we are interested in the waiting-time distributions and the queue length distributions (including any customer in service) at several specified points in time. Let $F_{i,j}$ be the number of customers present (waiting and in service) at Q_j when the server starts a visit period at Q_i (i.e. a polling epoch at Q_i). The joint distribution of $F_{i,1}$ and $F_{i,2}$ is represented by the two-dimensional probability generating function $\widetilde{F}_i(z_1, z_2) = \mathbb{E}[z_1^{F_{i,1}} z_2^{F_{i,2}}]$. Similarly, $\mathscr{F}_i$ represents the number of type-i customers present at a polling epoch of Q_i, provided that the previous visit period of the server was at Q_{3-i} and its probability generating function is given by $\widetilde{\mathscr{F}}_i(z) = \mathbb{E}[z^{\mathscr{F}_i}]$. The random variable L_j represents the number of customers at Q_j at an arbitrary point in time and the corresponding two-dimensional probability generating function is given by $\widetilde{L}(z_1, z_2) = \mathbb{E}[z_1^{L_1} z_2^{L_2}]$. The waiting time of a type-i customer that arrives at an arbitrary point in time is given by W_i, and its Laplace-Stieltjes transform is given by $\widetilde{W}_i(s) = \mathbb{E}[e^{-sW_i}]$.

We analyse the system under stability conditions ($\rho < 1$) and heavy-traffic conditions ($\rho \uparrow 1$). More specifically, in the latter regime, we scale the total arrival rate $\lambda_1 + \lambda_2$ while the ratio $\frac{\lambda_2}{\lambda_1}$ remains fixed. In this way, the heavy-traffic limit is uniquely defined.

It is moreover convenient, for any variable x that depends on the load ρ, to denote its value evaluated at $\rho = 1$ as $\hat{x}$. For example, $\hat{\rho}_i = \frac{\rho_i}{\rho}$, so that $\hat{\rho} = \hat{\rho}_1 + \hat{\rho}_2 = 1$ and $\hat{\lambda}_i = \frac{\hat{\rho}_i}{\mathbb{E}[B_i]}$. The waiting times and queue lengths tend to infinity in heavy traffic, and as a consequence their distributions are not well-defined in the limiting case. Therefore, we study the distributions of the scaled waiting times $\mathscr{W}_i = (1-\rho)W_i$ and the scaled queue lengths $\mathscr{L}_i = (1-\rho)L_i$. The Laplace-Stieltjes transform of the scaled waiting-time distribution is given by $\widetilde{\mathscr{W}}_i(s) = \mathbb{E}[e^{-s\mathscr{W}_i}]$. Likewise, the probability generating function of the scaled queue length distribution is given by $\widetilde{\mathscr{L}}_i(z) = \mathbb{E}[z^{\mathscr{L}_i}]$.

Finally, we will call any discrete random variable R to be geometrically (p) distributed if its probability mass function satisfies $\mathbb{P}(R = r) = (1-p)p^r$, and we use $\Sigma(\boldsymbol{z})$ throughout this chapter as shorthand notation for $\lambda_1(1-z_1) + \lambda_2(1-z_2)$.

7.3 Analysis for arbitrarily loaded systems

In this section, we derive explicit expressions for the marginal distributions of the waiting time in either queue and the joint queue length distribution. In Section 7.3.1, we first obtain expressions for $\widetilde{F}_i(z_1, z_2)$, the probability generating function corresponding to the joint queue length observed at a polling epoch of Q_i. These results ultimately lead in Section 7.3.2 to expressions for the quantities $\widetilde{W}_1(s)$, $\widetilde{W}_2(s)$ and $\widetilde{L}(z_1, z_2)$. Throughout this section, we assume that $\rho < 1$, i.e. the case where the queues are stable. In Section 7.4, we will study the limiting case $\rho \uparrow 1$, the case where the system becomes critically loaded.

7.3.1 Joint queue length at polling epochs

To obtain explicit expressions for the probability generating function $\widetilde{F}_i(z_1, z_2)$, we start with a functional equation for this function. Such a functional equation has already been derived in [271] for a setting consisting of multiple queues and a wide class of service disciplines. Applying these results to our case, we obtain

$$\widetilde{F}_1(z_1, z_2) = \nu_{1,1}\widetilde{F}_1(\widetilde{K}_{1,2}(z_2), z_2)\widetilde{M}_{1,1}(z_1, z_2) + \nu_{2,1}\widetilde{F}_2(z_1, \widetilde{K}_{2,1}(z_1))\widetilde{M}_{2,1}(z_1, z_2). \tag{7.2}$$

We will formally derive this functional equation in Section 8.3.1 under more general assumptions. For now, this equation can be seen to hold for the current model by the following observations. With probability $\nu_{i,1}$, a visit to Q_1 is preceded by a visit period at Q_i, during which each type-i customer initially present and all of its offspring is served (i.e. not only the customer himself, but also his children, grandchildren and so on). Over the course of each service of a type-i customer, a number of type-j customers, represented by the probability generating function $\widetilde{K}_{i,j}(z_j)$, arrives at Q_j. During the switch-over period $S_{i,1}$ between the two visits, the population of customers in the system grows with a number of arriving customers that is represented by $\widetilde{M}_{i,1}(z_1, z_2)$. By similar observations, we have that

$$\widetilde{F}_2(z_1, z_2) = \nu_{1,2}\widetilde{F}_1(\widetilde{K}_{1,2}(z_2), z_2)\widetilde{M}_{1,2}(z_1, z_2) + \nu_{2,2}\widetilde{F}_2(z_1, \widetilde{K}_{2,1}(z_1))\widetilde{M}_{2,2}(z_1, z_2). \tag{7.3}$$

We now develop explicit expressions for $\widetilde{F}_1(\widetilde{K}_{1,1}(z_2), z_2)$ and $\widetilde{F}_2(z_1, \widetilde{K}_{2,1}(z_1))$, so that (7.2) and (7.3) in turn offer explicit expressions for $\widetilde{F}_1(z_1, z_2)$ and $\widetilde{F}_2(z_1, z_2)$. To this end, we

note that substituting $z_1 = \widetilde{K}_{1,2}(z_2)$ in (7.2) leads to

$$\widetilde{F}_1(\widetilde{K}_{1,2}(z_2), z_2) = \frac{\nu_{2,1}\widetilde{M}_{2,1}(\widetilde{K}_{1,2}(z_2), z_2)}{1-\nu_{1,1}\widetilde{M}_{1,1}(\widetilde{K}_{1,2}(z_2), z_2)}\widetilde{F}_2(\widetilde{K}_{1,2}(z_2), \widetilde{K}_{2,1}(\widetilde{K}_{1,2}(z_2))). \tag{7.4}$$

Similarly, a substitution of $z_2 = \widetilde{K}_{2,1}(z_1)$ in (7.3) leads to

$$\widetilde{F}_2(z_1, \widetilde{K}_{2,1}(z_1)) = \frac{\nu_{1,2}\widetilde{M}_{1,2}(z_1, \widetilde{K}_{2,1}(z_1))}{1-\nu_{2,2}\widetilde{M}_{2,2}(z_1, \widetilde{K}_{2,1}(z_1))}\widetilde{F}_1(\widetilde{K}_{1,2}(\widetilde{K}_{2,1}(z_1)), \widetilde{K}_{2,1}(z_1)). \tag{7.5}$$

A combination of (7.4) and (7.5) gives

$$\widetilde{F}_1(\widetilde{K}_{1,2}(z_2), z_2) = a_1(z_2)\widetilde{F}_1(\widetilde{K}_{1,2}(b_1(z_2)), b_1(z_2)), \tag{7.6}$$

where

$$a_1(z_2) = \frac{\nu_{2,1}\widetilde{M}_{2,1}(\widetilde{K}_{1,2}(z_2), z_2)}{1-\nu_{1,1}\widetilde{M}_{1,1}(\widetilde{K}_{1,2}(z_2), z_2)}\,\frac{\nu_{1,2}\widetilde{M}_{1,2}(\widetilde{K}_{1,2}(z_2), b_1(z_2))}{1-\nu_{2,2}\widetilde{M}_{2,2}(\widetilde{K}_{1,2}(z_2), b_1(z_2))}$$

and

$$b_1(z_2) = \widetilde{K}_{2,1}(\widetilde{K}_{1,2}(z_2)).$$

Observe that (7.6) constitutes an expression for $\widetilde{F}_1(\widetilde{K}_{1,2}(z_2), \cdot)$ in terms of $\widetilde{F}_1(\widetilde{K}_{1,2}(z_2), \cdot)$ itself. Therefore, iteration of (7.6) leads to

$$\widetilde{F}_1(\widetilde{K}_{1,2}(z_2), z_2) = \widetilde{F}_1(\widetilde{K}_{1,2}(b_1^{(\infty)}(z_2)), b_1^{(\infty)}(z_2))\prod_{j=0}^{\infty} a_1(b_1^{(j)}(z_2)), \tag{7.7}$$

where $b_1^{(0)}(z_2) = z_2$ and $b_1^{(j)}(z_2) = b_1(b_1^{(j-1)}(z_2))$. By repeating the analysis above for $\widetilde{F}_2(z_1, \widetilde{K}_{2,1}(z_1))$, we obtain that

$$\widetilde{F}_2(z_1, \widetilde{K}_{2,1}(z_1)) = \widetilde{F}_2(b_2^{(\infty)}(z_1), \widetilde{K}_{2,1}(b_2^{(\infty)}(z_1)))\prod_{j=0}^{\infty} a_2(b_2^{(j)}(z_1)), \tag{7.8}$$

where

$$a_2(z_1) = \frac{\nu_{1,2}\widetilde{M}_{1,2}(z_1, \widetilde{K}_{2,1}(z_1))}{1-\nu_{2,2}\widetilde{M}_{2,2}(z_1, \widetilde{K}_{2,1}(z_1))}\,\frac{\nu_{2,1}\widetilde{M}_{2,1}(b_2(z_1), \widetilde{K}_{2,1}(z_1))}{1-\nu_{1,1}\widetilde{M}_{1,1}(b_2(z_1), \widetilde{K}_{2,1}(z_1))} \tag{7.9}$$

and

$$b_2(z_1) = \widetilde{K}_{1,2}(\widetilde{K}_{2,1}(z_1)),$$

$b_2^{(0)}(z_1) = z_1$ and $b_2^{(j)}(z_1) = b_2(b_2^{(j-1)}(z_1))$.

Now that explicit expressions for $\widetilde{F}_1(\widetilde{K}_{1,2}(z_2), z_2)$ and $\widetilde{F}_2(z_1, \widetilde{K}_{2,1}(z_1))$ are available, we show in the following two lemmas that the two terms $\widetilde{F}_1(\widetilde{K}_{1,2}(b_1^{(\infty)}(z_2)), b_1^{(\infty)}(z_2))$ and $\widetilde{F}_2(b_2^{(\infty)}(z_1), \widetilde{K}_{2,1}(b_2^{(\infty)}(z_1)))$ are well-defined constants and that the infinite products in (7.7) and (7.8) actually converge.

LEMMA 7.3.1. *For $z_1, z_2 \in \{z : z \in \mathbb{C} \wedge |z| \leq 1\}$, we have that $\widetilde{F}_1(\widetilde{K}_{1,2}(b_1^{(\infty)}(z_2)), b_1^{(\infty)}(z_2))$ and $\widetilde{F}_2(b_2^{(\infty)}(z_1), \widetilde{K}_{2,1}(b_2^{(\infty)}(z_1)))$ are well-defined constants equal to one.*

PROOF. See Appendix 7.A. □

LEMMA 7.3.2. *For $z_1, z_2 \in \{z : z \in \mathbb{C} \wedge |z| \leq 1\}$, the products $\prod_{j=0}^{\infty} a_1(b_1^{(j)}(z_2))$ and $\prod_{j=0}^{\infty} a_2(b_2^{(j)}(z_1))$ converge.*

PROOF. See Appendix 7.B. □

Now that we have analysed $\widetilde{F}_1(\widetilde{K}_{1,2}(z_2), z_2)$ and $\widetilde{F}_2(z_1, \widetilde{K}_{2,1}(z_1))$, we can derive expressions for $\widetilde{F}_1(z_1, z_2)$ and $\widetilde{F}_2(z_1, z_2)$ as follows.

THEOREM 7.3.3. *The probability generating functions $\widetilde{F}_1(z_1, z_2)$ and $\widetilde{F}_2(z_1, z_2)$, which correspond to the joint queue length at a polling epoch of Q_1 and Q_2, respectively, are given by*

$$\widetilde{F}_1(z_1, z_2) = \nu_{1,1}\widetilde{M}_{1,1}(z_1, z_2)\prod_{j=0}^{\infty} a_1(b_1^{(j)}(z_2)) + \nu_{2,1}\widetilde{M}_{2,1}(z_1, z_2)\prod_{j=0}^{\infty} a_2(b_2^{(j)}(z_1)) \tag{7.10}$$

and

$$\widetilde{F}_2(z_1, z_2) = \nu_{1,2}\widetilde{M}_{1,2}(z_1, z_2)\prod_{j=0}^{\infty} a_1(b_1^{(j)}(z_2)) + \nu_{2,2}\widetilde{M}_{2,2}(z_1, z_2)\prod_{j=0}^{\infty} a_2(b_2^{(j)}(z_1)). \tag{7.11}$$

PROOF. The theorems follows by combining (7.2), (7.3), (7.7), (7.8) with Lemmas 7.3.1 and 7.3.2. □

We use the expressions of Theorem 7.3.3 to obtain the (probability generating function of the) joint queue length distribution at an arbitrary point in time in Section 7.3.2. We conclude this section with a couple of remarks.

REMARK 7.3.1. The infinite products that arise in (7.10) and (7.11) have a clear interpretation. To see this, observe that by substituting $z_2 = 1$ in (7.10), one obtains $\widetilde{F}_1(z_1, 1) = \mathbb{E}[z_1^{F_{1,1}}]$, the probability generating function corresponding to the number of type-1 customers currently present at a polling epoch of Q_1. This yields

$$\widetilde{F}_1(z_1, 1) = \nu_{1,1}\widetilde{M}_{1,1}(z_1, 1) + \nu_{2,1}\widetilde{M}_{2,1}(z_1, 1)\prod_{j=0}^{\infty} a_2(b_2^{(j)}(z_1)), \tag{7.12}$$

since $a_1(1) = b_1(1) = 1$. This expression can be interpreted as follows. At the end of the previous visit period at Q_1, there are no type-1 customers in the system. Thus, with probability $\nu_{1,1}$, the number of type-1 customers that have arrived since the previous visit period at Q_1, did so over the course of a switch-over period $S_{1,1}$. This number of customers is represented by the probability generating function $\widetilde{M}_{1,1}(z_1, 1)$. With probability $\nu_{2,1}$, the previous visit period was at Q_2, so that $\widetilde{F}_1(z_1, 1)$ represents the probability generating function corresponding to $\mathscr{F}_1$ in this case, i.e. the number of type-1 customers present at a polling epoch of Q_1, given that the server's previous visit was at Q_2. This number of type-1 customers present not only consists of type-1 customers that arrived during a switch-over period $S_{2,1}$, but also type-1 customers that arrived between the end of the previous visit period at Q_1 and the end of the latest visit period at Q_2. As the former number of customers

is evidently represented by $\widetilde{M}_{2,1}(z_1,1)$, the infinite product $\prod_{j=0}^{\infty} a_2(b_2^{(j)}(z_1))$ should equal the probability generating function of the latter category of customers. From this, it also follows that $\widetilde{\mathscr{F}}_1(z) = \widetilde{M}_{2,1}(z,1)\prod_{j=0}^{\infty} a_2(b_2^{(j)}(z))$.

Another way to see that the infinite product $\prod_{j=0}^{\infty} a_2(b_2^{(j)}(z_1))$ represents the number of arriving type-1 customers between the last visit period end at Q_1 and subsequently the last visit period end at Q_2 is the following. Any type-1 customer currently present (i.e. at a polling epoch of Q_1) is a customer that either arrived during a switch-over period (an ancestor) or belongs to the offspring of another type-1 or type-2 customer that arrived during a switch-over period in the past (a descendant). The currently present type-1 customers that are (descendants of) ancestors that arrived during a particular period in the past are referred to as the contribution of that period to the current polling epoch. The expression $a_2(z_1)$ (cf. (7.9)) now represents the complete contribution of the period that lasted until the end of the last visit to Q_2 and started at the most recent visit to Q_2 before that time that directly preceded a Q_1 visit. This period starts with a switch-over period $S_{2,1}$, of which the contribution is easily seen to be given by $\widetilde{M}_{2,1}(b_2(z_1),\widetilde{K}_{2,1}(z_1))$. After that, a geometric number of switch-over periods from Q_1 to Q_1 occur, of which the (probability generating function of the) contribution is given by

$$\sum_{k=0}^{\infty} v_{2,1} v_{1,1}^k \widetilde{M}_{1,1}^k(b_2(z_1),\widetilde{K}_{2,1}(z_1)) = \frac{v_{2,1}}{1-v_{1,1}\widetilde{M}_{1,1}(b_2(z_1),\widetilde{K}_{2,1}(z_1))}.$$

Similarly, the contribution of the succeeding switch-over period $\widetilde{S}_{1,2}$ and the geometric number of switch-over periods from Q_2 to Q_2 are given by $\widetilde{M}_{1,2}(z_1,\widetilde{K}_{2,1}(z_1))$ and $\frac{v_{1,2}}{1-v_{2,2}\widetilde{M}_{2,2}(z_1,\widetilde{K}_{2,1}(z_1))}$, respectively. The product of these expressions constitutes $a_2(z_1) = a_2(b_2^{(0)}(z_1))$, the contribution of the latest 'inter visit-end period' of Q_2. Based on this, it is not hard to see, by the nature of $b_2(z_1)$, that $a_2(b_2^{(1)}(z_1))$ represents the contribution of the inter visit-end period preceding the latest inter visit-end period. Extending this observation, $a_2(b_2^{(j)}(z_1))$ represents the contribution of the j-th to last inter visit-end period of Q_2. As the customers currently present at Q_1 can be the contribution of any inter visit-end period of Q_2 in the past, the number sought is given by $\prod_{j=0}^{\infty} a_2(b_2^{(j)}(z_1))$, which represents the contribution of all inter visit-end periods that have past. An interpretation for $a_1(b_1^{(j)}(z_2))$ can be derived in a similar way.

REMARK 7.3.2. In the past, views similar to the contribution interpretation as presented in Remark 7.3.1 have led to numerical methods for several systems, such as the descendant set approach as developed in [145] for cyclic polling systems. It is shown there that by truncating the infinite products, accurate approximations of (the probability generating functions of) the marginal queue length distribution arise. This supports numerical observations that the infinite-product expressions as derived in this chapter give rise to efficient numerical means of computing queue length distributions.

7.3.2 Waiting time and joint queue length at an arbitrary point in time

Now that we have derived expressions for the probability generating function $\widetilde{F}_i(z_1,z_2)$ pertaining to the queue length at a polling epoch of Q_i, we use these results to obtain

$\widetilde{W}_i(s)$, the Laplace-Stieltjes transform of the waiting-time distribution of type-i customers, and $\widetilde{L}(z_1,z_2)$, the probability generating function representing the joint queue length distribution at an arbitrary point in time.

7.3.2.1 Analysis of $\widetilde{W}_i(s)$

To extract an expression for $\widetilde{W}_i(s)$ from the expressions found in Section 7.3.1, we use the observation given in [271, pp. 90–91] that the analysis found in [233, Section 4.3] applied to Markovian polling systems leads to

$$\widetilde{W}_1(\lambda_1(1-z)) = \frac{q_1(1-\rho)(1-\widetilde{F}_1(z,1))}{\sigma\lambda_1(\widetilde{B}_1(\lambda_1(1-z))-z)} \tag{7.13}$$

and

$$\widetilde{W}_2(\lambda_2(1-z)) = \frac{q_2(1-\rho)(1-\widetilde{F}_2(1,z))}{\sigma\lambda_2(\widetilde{B}_2(\lambda_2(1-z))-z)}, \tag{7.14}$$

where σ, as defined in Section 7.2, denotes the average duration of an arbitrary switchover period. This observation leads to expressions for $\widetilde{W}_i(s)$ as stated in the following theorem.

THEOREM 7.3.4. *The Laplace-Stieltjes transform of the waiting-time distribution of type-j customers is given by*

$$\widetilde{W}_j(s) = \frac{q_j(1-\rho)}{\sigma(s-\lambda_j(1-\widetilde{B}_j(s)))} \times \left(1-\sum_{i=1}^{2} v_{i,j}\widetilde{S}_{i,j}(s)\left(\mathbb{1}_{\{i=j\}}+\mathbb{1}_{\{i\neq j\}}\prod_{k=0}^{\infty}a_i\left(b_i^{(k)}\left(1-\frac{s}{\lambda_j}\right)\right)\right)\right).$$

PROOF. By substituting $s=\lambda_1(1-z)$ and $s=\lambda_2(1-z)$, respectively, in (7.13) and (7.14), we obtain

$$\widetilde{W}_1(s) = \frac{q_1(1-\rho)(1-\widetilde{F}_1(1-\frac{s}{\lambda_1},1))}{\sigma(s-\lambda_1(1-\widetilde{B}_1(s)))} \tag{7.15}$$

and

$$\widetilde{W}_2(s) = \frac{q_2(1-\rho)(1-\widetilde{F}_2(1,1-\frac{s}{\lambda_2}))}{\sigma(s-\lambda_2(1-\widetilde{B}_2(s)))}. \tag{7.16}$$

Combining these expressions with (7.12) and its equivalent for $\widetilde{F}_2(1,z_2)$ leads to the theorem. □

7.3.2.2 Analysis of $\widetilde{L}(z_1,z_2)$

To obtain $\widetilde{L}(z_1,z_2)$, we use an approach that is introduced in [51]. Before we derive the probability generating function corresponding to the joint queue length at an arbitrary

point in time, we first regard $\widetilde{X}_i(z_1,z_2)=\mathbb{E}[z_1^{X_{i,1}}z_2^{X_{i,2}}]$, the probability generating function representing the queue lengths $X_{i,1}$ and $X_{i,2}$ of Q_1 and Q_2 at an arbitrary point during a visit period at Q_i. It turns out to hold that

$$\widetilde{X}_1(z_1,z_2)=\frac{q_1(1-\rho)}{\rho_1\sigma}\frac{z_1(\widetilde{F}_1(z_1,z_2)-\widetilde{F}_1(\widetilde{K}_{1,2}(z_2),z_2))}{z_1-\widetilde{B}_1(\Sigma(z))}\frac{1-\widetilde{B}_1(\Sigma(z))}{\Sigma(z)} \tag{7.17}$$

and

$$\widetilde{X}_2(z_1,z_2)=\frac{q_2(1-\rho)}{\rho_2\sigma}\frac{z_2(\widetilde{F}_2(z_1,z_2)-\widetilde{F}_2(z_1,\widetilde{K}_{2,1}(z_1)))}{z_2-\widetilde{B}_2(\Sigma(z))}\frac{1-\widetilde{B}_2(\Sigma(z))}{\Sigma(z)}. \tag{7.18}$$

We will formally derive these results in Section 8.4 under more general assumptions (i.e. not necessarily two queues or exhaustive service). Furthermore, the results of Section 8.4 reveal that $\widetilde{Y}_{i,j}(z_1,z_2)=\mathbb{E}[z_1^{Y_{i,j,1}}z_2^{Y_{i,j,2}}]$, the probability generating function representing the queue lengths $Y_{i,j,1}$ and $Y_{i,j,2}$ of Q_1 and Q_2 at an arbitrary point during a switch-over period from Q_i to Q_j is given by

$$\widetilde{Y}_{1,j}(z_1,z_2)=\widetilde{F}_1(\widetilde{K}_{1,2}(z_2),z_2)\frac{1-\widetilde{M}_{1,j}(z_1,z_2)}{\Sigma(z)\mathbb{E}[S_{1,j}]} \tag{7.19}$$

and

$$\widetilde{Y}_{2,j}(z_1,z_2)=\widetilde{F}_2(z_1,\widetilde{K}_{2,1}(z_1))\frac{1-\widetilde{M}_{2,j}(z_1,z_2)}{\Sigma(z)\mathbb{E}[S_{2,j}]}. \tag{7.20}$$

We now combine the expressions (7.17)–(7.20) into one expression for $\widetilde{L}(z_1,z_2)$, the probability generating function representing the joint queue length at an arbitrary point in time. Observe that the server serves Q_i a fraction ρ_i of the time. In the remaining fraction $1-\rho$ of the time, the server is setting up for service at another queue. Of the time the server is in a switch-over period, he spends a fraction $\frac{\nu_{i,j}q_j\mathbb{E}[S_{i,j}]}{\sigma}$ setting up from Q_i to Q_j. Therefore, we have that

$$\widetilde{L}(z_1,z_2)=\sum_{i=1}^{2}\left(\rho_i\widetilde{X}_i(z_1,z_2)+\frac{1-\rho}{\sigma}\sum_{j=1}^{2}\nu_{i,j}q_j\mathbb{E}[S_{i,j}]\widetilde{Y}_{i,j}(z_1,z_2)\right). \tag{7.21}$$

This leads to the following theorem.

THEOREM 7.3.5. *The probability generating function of the joint queue length distribution is given by*

$$\widetilde{L}(z_1,z_2)=\frac{1-\rho}{\Sigma(z)\sigma}\sum_{i=1}^{2}\sum_{j=1}^{2}q_j\Bigg(\frac{z_j(1-\widetilde{B}_j(\Sigma(z)))}{z_j-\widetilde{B}_j(\Sigma(z))}(\nu_{i,j}\widetilde{M}_{i,j}(z_1,z_2)-\mathbb{1}_{\{i=j\}})$$
$$+\nu_{i,j}(1-\widetilde{M}_{i,j}(z_1,z_2))\Bigg)\prod_{k=0}^{\infty}a_i(b_i^{(k)}(z_{3-i})).$$

PROOF. The theorem follows by combining (7.7), (7.8), Lemma 7.3.1 and Theorem 7.3.3 with (7.17)–(7.21). □

7.4 Heavy-traffic asymptotics

In Section 7.3, we derived expressions for the Laplace-Stieltjes transforms of the waiting-time distributions and the probability generating function of the joint queue length distribution. These expressions are suitable for computational purposes, as theoretical and numerical evidence shows that the infinite products contained in these expressions converge very fast. However, the expressions are not in closed form, and the probability generating functions and the Laplace-Stieltjes transforms found are hard to invert. In an effort to obtain closed-form expressions for the distributions themselves, we consider the heavy-traffic asymptotics of the system, i.e. the behaviour of the system when $\rho \uparrow 1$. Recall that we study the case where the heavy-traffic limit $\rho \uparrow 1$ is taken by scaling the total arrival rate $\lambda_1+\lambda_2$ such that the ratio $\frac{\lambda_2}{\lambda_1}$ remains fixed, so that $\frac{\hat{\lambda}_2}{\hat{\lambda}_1} = \frac{\lambda_2}{\lambda_1}$, with $\hat{\lambda}_i$ as defined in Section 7.2. In this regime, the waiting times and the queue lengths tend to infinity. Therefore, we now study the scaled waiting times $\mathscr{W}_i$ as well as the scaled queue lengths $\mathscr{L}_i$ and obtain closed-form expressions directly for their distributions. These expressions are not only easy to implement, but they also give insight into the primary effects of the model parameters on the waiting times and queue lengths when the system operates under a heavy load. In Section 7.4.1, we derive the heavy-traffic behaviour of the waiting times and queue lengths incurred by the customers based on previous results for cyclic polling systems and some insightful observations. Subsequently, we rigorously prove these results in Section 7.4.2.

7.4.1 Initial study of the heavy-traffic behaviour

Before we study the heavy-traffic behaviour of the model in its full generality, we first consider the degenerate case $\xi_1 = \xi_2 = 0$ of our model. Note that for $\xi_1 = \xi_2 = 0$, the server always switches from Q_1 to Q_2 or from Q_2 to Q_1. Thus, in this particular case, the server follows a fixed alternating (or cyclic) routing mechanism. The heavy-traffic behaviour of cyclic polling models that are of a branching type and consist of an arbitrary number of queues has already been established in [184, 248, 249]. Translating this to our setting with two queues, exhaustive service and cyclic routing ($\xi_1 = \xi_2 = 0$), these results readily imply the following.

PROPOSITION 7.4.1. *For $\xi_1 = \xi_2 = 0$, the Laplace-Stieltjes transform of the limiting scaled waiting-time distribution is, in the heavy-traffic regime, given by*

$$\lim_{\rho\uparrow 1} \widetilde{\mathscr{W}_i}(s) = \frac{1}{s(1-\hat{\rho}_i)(\mathbb{E}[S_{1,2}]+\mathbb{E}[S_{2,1}])} \left(1 - \left(\frac{\mu_i^{cyc}}{\mu_i^{cyc}+s}\right)^{\alpha^{cyc}}\right),$$

where

$$\alpha^{cyc} = \frac{2\hat{\rho}_1\hat{\rho}_2(\mathbb{E}[S_{1,2}]+\mathbb{E}[S_{2,1}])}{\hat{\lambda}_1\mathbb{E}[B_1^2]+\hat{\lambda}_2\mathbb{E}[B_2^2]} \text{ and } \mu_i^{cyc} = \frac{2\hat{\rho}_i}{\hat{\lambda}_1\mathbb{E}[B_1^2]+\hat{\lambda}_2\mathbb{E}[B_2^2]}.$$

Equivalently,

$$\lim_{\rho\uparrow 1} \mathbb{P}(\mathscr{W}_i \le t) = \mathbb{P}(UI_i \le t),$$

where U is a uniformly $[0,1]$ distributed random variable, I_i is a gamma distributed random variable with shape parameter $\alpha^{cyc}+1$ and scale parameter μ_i^{cyc}, and U and I_i are independent.

The given distribution function immediately follows from inversion of the limiting Laplace-Stieltjes transform. We observe that for the cyclic system, the complete heavy-traffic distribution of the waiting time only depends on the switch-over times through their first moments. In fact, the scaled waiting-time distribution only depends on the complete switch-over time distributions $S_{1,2}$ and $S_{2,1}$ through $\mathbb{E}[S_{1,2}] + \mathbb{E}[S_{2,1}]$, the *first* moment of the *total* switch-over time incurred between two polling epochs at Q_1.

Next, we observe for the general case (i.e. $0 \leq \xi_1, \xi_2 < 1$) the following. A period between two polling epochs at Q_1 can be divided in a number of subperiods:

(i) The first visit period at Q_1 after having visited Q_2;

(ii) A geometric (ξ_1) number of switch-over periods from Q_1 to Q_1 and subsequent 'revisit' periods at Q_1;

(iii) The switch-over period from Q_1 to Q_2;

(iv) The first visit period at Q_2 after having visited Q_1;

(v) A geometric (ξ_2) number of switch-over periods from Q_2 to Q_2 and subsequent 'revisit' periods at Q_2;

(vi) The switch-over period from Q_2 to Q_1.

In this view, we can draw a connection between the general case and the cyclic polling model as described above by slightly adjusting this order of events as follows:

(a) All visit periods between a polling epoch at Q_1 and the first polling epoch at Q_2 to occur afterwards;

(b) A geometric (ξ_1) number of switch-over periods from Q_1 to Q_1;

(c) The switch-over period from Q_1 to Q_2;

(d) All visit periods between the polling epoch at Q_2 and the first polling epoch at Q_1 to occur afterwards;

(e) A geometric (ξ_2) number of switch-over periods from Q_2 to Q_2;

(f) The switch-over period from Q_2 to Q_1.

Thus, the 'revisit' periods from the subperiods (ii) and (v) have been shifted to the subperiods (a) and (d). In the heavy-traffic regime, the implications of this adjustment are, however, negligible. This is the case because the additional customers served in the subperiods (a) and (d) with respect to those in the original subperiods (i) and (iv) are finite in number (they constitute arrivals during finitely long switch-over times). However, since these 'original customers' are infinite in number in the heavy-traffic regime, the finite number of additional customers scales away in heavy traffic. As a result, the limiting waiting-time distribution of the customers served in the periods (i) and (iv) coincides in the heavy-traffic regime with that of the customers served in the reordered subperiods (a) and (d), respectively. Note that in this reordered scheme, the polling system can be interpreted as a cyclic model, as the subperiods (b) and (c) together form a switch-over period from Q_1 to Q_2, and the subperiods (e) and (f) together form a switch-over period from Q_2 to Q_1. The switch-over period from Q_1 to Q_2 in this cyclic equivalent then consists of a geometric (ξ_1) number of original switch-over periods from Q_1 to Q_1 and an

original switch-over period from Q_1 to Q_2 of the Markovian model. Similarly, the switch-over period from Q_2 to Q_1 in the cyclic equivalent consists of a geometric (ξ_2) number of switch-over periods from Q_2 to Q_2 and a subsequent switch-over period from Q_2 to Q_1.

Finally, we note that the first moment of the total switch-over time incurred between two polling epochs at Q_1, which we denote by $\mathbb{E}[S^{tot}]$, is in our case given by

$$\begin{aligned}\mathbb{E}[S^{tot}] &= \sum_{i=0}^{\infty}\sum_{j=0}^{\infty}(i\mathbb{E}[S_{1,1}]+\mathbb{E}[S_{1,2}]+\mathbb{E}[S_{2,1}]+j\mathbb{E}[S_{2,2}])(1-\xi_1)\xi_1^i(1-\xi_2)\xi_2^j \\ &= \frac{\xi_1}{1-\xi_1}\mathbb{E}[S_{1,1}]+\mathbb{E}[S_{1,2}]+\mathbb{E}[S_{2,1}]+\frac{\xi_2}{1-\xi_2}\mathbb{E}[S_{2,2}]. \qquad (7.22)\end{aligned}$$

Combining all of the observations above, it is easily understood that the heavy-traffic behaviour of the general case is similar to the heavy-traffic behaviour as derived in Proposition 7.4.1 for the cyclic case, except that the term $\mathbb{E}[S_{1,2}]+\mathbb{E}[S_{2,1}]$ should be replaced by $\mathbb{E}[S^{tot}]$. We formulate this result below. A rigorous proof will be given in Section 7.4.2.

THEOREM 7.4.2. *For $0 \le \xi_1, \xi_2 < 1$, the Laplace-Stieltjes transform of the limiting scaled waiting-time distribution is given by*

$$\lim_{\rho\uparrow 1}\widetilde{\mathscr{W}_i}(s) = \frac{1}{s(1-\hat{\rho}_i)\mathbb{E}[S^{tot}]}\left(1-\left(\frac{\mu_i}{\mu_i+s}\right)^{\alpha}\right), \qquad (7.23)$$

where

$$\alpha = \frac{2\hat{\rho}_1\hat{\rho}_2\mathbb{E}[S^{tot}]}{\hat{\lambda}_1\mathbb{E}[B_1^2]+\hat{\lambda}_2\mathbb{E}[B_2^2]},\ \mu_i = \frac{2\hat{\rho}_i}{\hat{\lambda}_1\mathbb{E}[B_1^2]+\hat{\lambda}_2\mathbb{E}[B_2^2]} \qquad (7.24)$$

and $\mathbb{E}[S^{tot}]$ is given in (7.22)*. Equivalently,*

$$\lim_{\rho\uparrow 1}\mathbb{P}(\mathscr{W}_i \le t) = \mathbb{P}(UI_i \le t), \qquad (7.25)$$

where U is a uniformly $[0,1]$ distributed random variable, I_i is a gamma distributed random variable with shape parameter $\alpha+1$ and scale parameter μ_i, and U and I_i are independent.

Based on this theorem concerning the scaled waiting-time distribution, we can also derive the heavy-traffic distribution of the scaled queue length distribution. From Little's law, it is immediate that $\mathbb{E}[\mathscr{L}_i] = \hat{\lambda}_i\mathbb{E}[\mathscr{W}_i]$. Furthermore, in many queueing models under heavy-traffic conditions, the scaled virtual waiting-time processes and queue length processes exhibit so-called state-space collapse (cf. [205]), similar to what we encountered in Section 3 for the extended machine repair model. It is thus reasonable to assume that in heavy traffic the distribution of $\mathscr{L}_i$ equals the distribution of $\mathscr{W}_i$ scaled by a factor $\hat{\lambda}_i$. This leads to the following statement, for which again a rigorous proof will be given in Section 7.4.2.

THEOREM 7.4.3. *For $0 \le \xi_1, \xi_2 < 1$, the limiting scaled marginal queue length distribution is given by*

$$\lim_{\rho\uparrow 1}\mathbb{P}(\mathscr{L}_i \le t) = \mathbb{P}(UI_i \le t),$$

where U is a uniformly $[0,1]$ distributed random variable and I_i is a gamma distributed random variable with shape parameter $\alpha+1$ and scale parameter $\frac{\mu_i}{\hat{\lambda}_i}$ (α and μ_i as defined in (7.24)*). Furthermore, the random variables U and I_i are independent.*

REMARK 7.4.1. Besides the distribution of a uniform times a gamma random variable, the limiting distribution of $(1-\rho)W_i$ as given in Theorem 7.4.2 can also be interpreted as the distribution of the residual (overshoot) of a gamma distributed random variable. To see this, observe that (7.23) can be rewritten as

$$\lim_{\rho\uparrow 1} \widetilde{W}_i(s) = \frac{1-\left(\frac{\mu_i}{\mu_i+s}\right)^{\alpha}}{s\frac{\alpha}{\mu_i}}.$$

As $(\frac{\mu_i}{\mu_i+s})^{\alpha}$ is the Laplace-Stieltjes transform of a gamma (α,μ_i) distribution with first moment $\frac{\alpha}{\mu_i}$, the limiting distribution of the scaled waiting time represents the distribution of the residual (overshoot) of a gamma distributed random variable with shape parameter α and scale parameter μ_i. A similar observation holds for the limiting distribution of $(1-\rho)W_i$ in the cyclic case as provided in Proposition 7.4.1.

REMARK 7.4.2. Theorems 7.4.2 and 7.4.3 can immediately be used as approximations for the marginal waiting-time distributions and queue length distributions in stable systems with a load $\rho < 1$:

$$\mathbb{P}(W_i < t) \approx \mathbb{P}(UI_i < (1-\rho)t) \text{ and } \mathbb{P}(L_i < x) \approx \mathbb{P}\left(UI_i < \hat{\lambda}_i(1-\rho)\left(x-\frac{1}{2}\right)\right),$$

where U and I_i are independent random variables with a uniform $[0,1]$ distribution and a gamma $(\alpha+1,\mu_i)$ distribution, respectively. Furthermore, the parameters α and μ_i are as defined in (7.24), and the term $1/2$ in the right-hand side of the second approximation appears for reasons of continuity correction. As shown in [184], approximations of this type are reasonably accurate for heavily loaded polling models (i.e. a load close to one). This is not surprising, as these approximations have the correct heavy-traffic limiting behaviour by construction. Moreover, it is interesting to note that the limiting distributions of the scaled waiting times and queue lengths only depend on the first two moments of the service time distribution as well as the first moment of the total switch-over time between two polling epochs at Q_1. They do not require higher moments and are thus useful for practical purposes, since in reality, information about third-order and higher-order moments is often hard to get. When one is interested in approximations that also perform well for lightly loaded systems, one may refine the approximations in the spirit of [P9, 45] or Section 4.3. More specifically, one may consider to construct approximations by interpolating between the found known light-traffic behaviour and heavy-traffic asymptotics based on the actual load offered to the system.

7.4.2 Proofs of Theorems 7.4.2 and 7.4.3

In this section, we prove Theorems 7.4.2 and 7.4.3. For the former theorem, we rely in part on the results found in [184]. This paper provides an analysis of the heavy-traffic behaviour of periodic polling systems of which the marginal queue length distribution at polling epochs can be (numerically) computed by the descendant set approach (cf. [145]). More specifically, [184] studies the heavy-traffic behaviour of these systems by analysing the mechanics of this technique in the heavy-traffic regime. The results that we particularly rely on are [184, Theorems 3 and 4], which give the limiting behaviour of

the marginal queue length Ξ of Q_1 observed at predefined epochs in time, of which the corresponding probability generating function $\widetilde{\Xi}(z) = \mathbb{E}[z^{\Xi}]$ can be written as

$$\widetilde{\Xi}(z) = \prod_{c=0}^{\infty} \widetilde{R}_1\left(\lambda_1(1-\widetilde{A}_{1,c-1}(z)) + \lambda_2(1-\widetilde{A}_{2,c}(z))\right) \times \widetilde{R}_2\left(\lambda_1(1-\widetilde{A}_{1,c-1}(z)) + \lambda_2(1-\widetilde{A}_{2,c-1}(z))\right), \tag{7.26}$$

where $\widetilde{R}_1(s)$ and $\widetilde{R}_2(s)$ are Laplace-Stieltjes transforms of two positive random variables R_1 and R_2,

$$\begin{aligned} \widetilde{A}_{1,c}(z) &= \widetilde{\Gamma}_1(\lambda_2(1-\widetilde{A}_{2,c}(z))) = \widetilde{K}_{1,2}(\widetilde{A}_{2,c}(z)), & \widetilde{A}_{1,-1}(z) &= z, \\ \widetilde{A}_{2,c}(z) &= \widetilde{\Gamma}_2(\lambda_1(1-\widetilde{A}_{1,c-1}(z))) = \widetilde{K}_{2,1}(\widetilde{A}_{1,c-1}(z)), & \widetilde{A}_{2,-1}(z) &= 1 \end{aligned} \tag{7.27}$$

and $\widetilde{\Gamma}_i(s)$ is as defined in Section 7.2. The results of [184] state that under these conditions, $(1-\rho)\Xi$ converges in distribution, as $\rho \uparrow 1$, to a gamma distributed random variable with shape parameter $\frac{2\hat{\rho}_1\hat{\rho}_2(\mathbb{E}[R_1]+\mathbb{E}[R_2])}{\hat{\lambda}_1\mathbb{E}[B_1^2]+\hat{\lambda}_2\mathbb{E}[B_2^2]}$ and scale parameter $\frac{2\hat{\rho}_1}{\hat{\lambda}_1(\hat{\lambda}_1\mathbb{E}[B_1^2]+\hat{\lambda}_2\mathbb{E}[B_2^2])}$. Furthermore, it is stated that $\lim_{\rho\uparrow 1}\mathbb{E}[(1-\rho)^k\Xi^k]$ coincides with the k-th moment of this distribution.

We have now only stated the results of [184] applied to two-queue polling systems with alternating and exhaustive service. A more general statement for polling systems with a general number of queues and periodic routing is shown to hold in [184] by exploiting several useful observations based on the descendant set approach.

As noted in Remark 7.3.2, however, the expressions that we obtained for the probability generating function of the queue length distribution in Section 7.3 allow for an interpretation in the spirit of the descendant set approach. As a result, the results of [184] as stated above almost directly lead to the following lemma pertaining to $\mathscr{F}_i$, the number of type-i customers in the system at a polling epoch of Q_i that follows a visit period at Q_{3-i}.

LEMMA 7.4.4. *The distribution of $(1-\rho)\mathscr{F}_i$ converges, as $\rho \uparrow 1$, to a gamma distribution with shape parameter α and scale parameter $\mu_i/\hat{\lambda}_i$, where α and μ_i are defined in* (7.24). *Furthermore, we have that $\lim_{\rho\uparrow 1}\mathbb{E}[(1-\rho)^k\mathscr{F}_i^k]$ coincides with the k-th moment of this distribution.*

PROOF. We focus on the limiting distribution of $(1-\rho)\mathscr{F}_1$. In Remark 7.3.1, we already concluded that $\widetilde{\mathscr{F}}_1(z) = \widetilde{M}_{2,1}(z,1)\prod_{j=0}^{\infty} a_2(b_2^{(j)}(z))$. With some effort, it is straightforward to see that this can be written alternatively as

$$\widetilde{\mathscr{F}}_1(z) = \widetilde{\Xi}(z)\frac{1-\nu_{1,1}\widetilde{M}_{1,1}(z,1)}{\nu_{2,1}\widetilde{M}_{2,1}(z,1)}, \tag{7.28}$$

where $\widetilde{\Xi}(z)$ is defined as in (7.26) with

$$\widetilde{R}_j(s) = \widetilde{S}_{j,3-j}(s)\frac{\nu_{j,3-j}}{1-\nu_{3-j,3-j}\widetilde{S}_{3-j,3-j}(s)},$$

i.e. R_j is chosen to be the sum of a switch-over time from Q_j to Q_{3-j} and an independent geometric $(\nu_{3-j,3-j})$ number of independent switch-over times from Q_{3-j} to Q_{3-j}. From

this definition, it is easily verified that $\mathbb{E}[R_1]+\mathbb{E}[R_2]=\mathbb{E}[S^{tot}]$. As $\lim_{\rho\uparrow 1}\widetilde{M}_{1,1}(z^{1-\rho},1)=\lim_{\rho\uparrow 1}\widetilde{M}_{2,1}(z^{1-\rho},1)=1$, it is clear from (7.28) that the probability generating function of the scaled distribution $\widetilde{\mathscr{F}}_1(z^{1-\rho})=\mathbb{E}[z^{(1-\rho)\mathscr{F}_1}]$ satisfies

$$\lim_{\rho\uparrow 1}\widetilde{\mathscr{F}}_1(z^{1-\rho})=\lim_{\rho\uparrow 1}\widetilde{\Xi}(z^{1-\rho}).$$

Thus, the distributions of the scaled versions of $\mathscr{F}_1$ and Ξ coincide in the heavy-traffic limit due to Lévy's continuity theorem (cf. [277, Section 18.1]), which connects pointwise convergence of Laplace-Stieltjes transforms with convergence in distribution. For $i=1$, the lemma now follows from the results of [184] as described above. For $i=2$, the lemma follows by interchanging indices. □

Now that we have established the heavy-traffic behaviour of $\mathscr{F}_i$, we are able to prove Theorem 7.4.2 by making use of (7.15) and (7.16).

PROOF OF THEOREM 7.4.2. Again, we focus on the case $i=1$ with the understanding that the proof for the case $i=2$ follows by interchanging indices. By (7.12) and (7.15), we have that

$$\begin{aligned}\lim_{\rho\uparrow 1}\widetilde{\mathscr{W}}_1(s)=&\lim_{\rho\uparrow 1}\frac{q_1(1-\rho)}{\sigma((1-\rho)s-\lambda_1(1-\widetilde{B}_1((1-\rho)s)))}\\&\times\lim_{\rho\uparrow 1}\left(1-\nu_{1,1}\widetilde{M}_{1,1}\left(1-\frac{(1-\rho)s}{\lambda_1},1\right)-\nu_{2,1}\widetilde{\mathscr{F}}_1\left(1-\frac{(1-\rho)s}{\lambda_1}\right)\right).\end{aligned}\tag{7.29}$$

By applying L'Hôpital's rule and observing that $\frac{q_1}{\sigma}=(\nu_{2,1}\mathbb{E}[S^{tot}])^{-1}$, we obtain for the first term in the right-hand side that

$$\begin{aligned}&\lim_{\rho\uparrow 1}\frac{q_1(1-\rho)}{\sigma((1-\rho)s-\lambda_1(1-\widetilde{B}_1((1-\rho)s)))}\\&=\lim_{\rho\uparrow 1}\frac{-q_1}{\sigma s(-1+\lambda_1\mathbb{E}[B_1e^{-(1-\rho)sB_1}])}=\frac{1}{\nu_{2,1}s(1-\hat{\rho}_1)\mathbb{E}[S^{tot}]}.\end{aligned}$$

Furthermore, it is clear that $\lim_{\rho\uparrow 1}\widetilde{M}_{1,1}(1-\frac{(1-\rho)s}{\lambda_1},1)=1$. Deriving $\lim_{\rho\uparrow 1}\widetilde{\mathscr{F}}_1(1-\frac{(1-\rho)s}{\lambda_1})$ takes a bit more effort. By invoking a Taylor expansion in $\mathscr{F}_1$, we have that

$$\lim_{\rho\uparrow 1}\widetilde{\mathscr{F}}_1\left(1-\frac{(1-\rho)s}{\lambda_1}\right)=\lim_{\rho\uparrow 1}\mathbb{E}\left[\left(1-\frac{(1-\rho)s}{\lambda_1}\right)^{\mathscr{F}_1}\right]=\lim_{\rho\uparrow 1}\mathbb{E}\left[\sum_{k=0}^{\infty}\frac{\log^k(1-\frac{(1-\rho)s}{\lambda_1})\mathscr{F}_1^k}{k!}\right].$$

To further reduce this expression, observe that a Taylor expansion around $\rho=1$ yields $\log(1-(1-\rho)c)=-\sum_{j=1}^{\infty}\frac{(1-\rho)^jc^j}{j}$ for any $c\in\mathbb{R}$. Hence,

$$\lim_{\rho\uparrow 1}\widetilde{\mathscr{F}}_1\left(1-\frac{(1-\rho)s}{\lambda_1}\right)=\lim_{\rho\uparrow 1}\mathbb{E}\left[\sum_{k=0}^{\infty}\frac{(-1)^k\left(\sum_{j=1}^{\infty}(1-\rho)^js^j\lambda_1^{-j}/j\right)^k\mathscr{F}_1^k}{k!}\right].\tag{7.30}$$

Note, however, that due to Lemma 7.4.4, we have for any $j > k$ that $\lim_{\rho\uparrow 1}\mathbb{E}[(1-\rho)^j\mathscr{F}_1^k] = \lim_{\rho\uparrow 1}(1-\rho)^{j-k}\lim_{\rho\uparrow 1}\mathbb{E}[(1-\rho)^k\mathscr{F}_1^k] = 0$. Therefore, second-order and higher-order terms of the inner sum of (7.30) disappear in the limit, so that the expression as a whole reduces to

$$\begin{aligned}\lim_{\rho\uparrow 1}\widetilde{\mathscr{F}}_1\left(1-\frac{(1-\rho)s}{\lambda_1}\right) &= \lim_{\rho\uparrow 1}\mathbb{E}\left[\sum_{k=0}^{\infty}\frac{(-1)^k(1-\rho)^k s^k\lambda_1^{-k}\mathscr{F}_1^k}{k!}\right]\\ &= \lim_{\rho\uparrow 1}\mathbb{E}[e^{-(1-\rho)\frac{s}{\lambda_1}\mathscr{F}_1}] = \left(\frac{\mu_1}{\mu_1+s}\right)^{\alpha},\end{aligned}$$

where the last equality follows from Lemma 7.4.4. By combining the limits found above, we can reduce (7.29) to

$$\lim_{\rho\uparrow 1}\widetilde{\mathscr{W}}_1(s) = \frac{1}{\nu_{2,1}s(1-\hat{\rho}_1)\mathbb{E}[S^{tot}]}\left(1-\nu_{1,1}-\nu_{2,1}\left(\frac{\mu_1}{\mu_1+s}\right)^{\alpha}\right),$$

which is equivalent to (7.23). Equation (7.25) now follows by inversion of the Laplace-Stieltjes transform and the subsequent use of Lévy's continuity theorem. □

Now that Theorem 7.4.2 is proved, Theorem 7.4.3 follows almost immediately by the proof below.

PROOF OF THEOREM 7.4.3. We make use of the distributional form of Little's law (cf. [135]), which states that

$$\widetilde{L}_i(z) = \widetilde{W}_i(\lambda_i(1-z))\widetilde{B}_i(\lambda_i(1-z)).$$

Consequently, we have that

$$\begin{aligned}\lim_{\rho\uparrow 1}\widetilde{\mathscr{L}}_i(z) &= \lim_{\rho\uparrow 1}\widetilde{L}_i(z^{1-\rho}) = \lim_{\rho\uparrow 1}\widetilde{W}_i(\lambda_i(1-z^{1-\rho}))\widetilde{B}_i(\lambda_i(1-z^{1-\rho}))\\ &= \lim_{\rho\uparrow 1}\widetilde{\mathscr{W}}_i\left(\frac{\lambda_i(1-z^{1-\rho})}{1-\rho}\right). \qquad (7.31)\end{aligned}$$

As $\lim_{\rho\uparrow 1}\frac{\lambda_i(1-z^{1-\rho})}{1-\rho} = -\hat{\lambda}_i\log(z)$, a combination of Theorem 7.4.2 and (7.31) now implies that

$$\lim_{\rho\uparrow 1}\widetilde{\mathscr{L}}_i(z) = \frac{1}{-\hat{\lambda}_i\log(z)(1-\hat{\rho}_i)\mathbb{E}[S^{tot}]}\left(1-\left(\frac{\mu_i}{\mu_i-\hat{\lambda}_i\log(z)}\right)^{\alpha}\right).$$

The latter expression is the probability generating function of the distribution mentioned in the theorem. A straightforward application of Lévy's continuity theorem thus concludes the proof. □

REMARK 7.4.3. The striking similarity between the heavy-traffic asymptotics of cyclic polling systems and those of the class of systems that we consider may in part be explained by the following. Despite the fact that Markovian polling systems generally do not satisfy the branching property as introduced in Section 1.3.2, the subset of two-queue exhaustive models does actually satisfy this property. More specifically, in the model that we consider in this chapter, the joint queue length process observed at Q_i polling epochs

constitutes a multi-type branching process with immigration (see e.g. [21]). As a consequence, this model fits in the framework considered in [249], and Lemma 7.4.4 follows alternatively from [249, Theorem 5] by taking the particle offspring functions $f^{(i)}(z_1,z_2)$ and the immigration function $g(z_1,z_2)$ as introduced in [249, Equations (3) and (4)] equal to

$$f^{(1)}(z_1,z_2)=\widetilde{K}_{1,2}(\widetilde{K}_{2,1}(z_1)),\ \ f^{(2)}(z_1,z_2)=\widetilde{K}_{2,1}(z_1)$$

and

$$g(z_1,z_2)=a_2(z_1)\frac{\widetilde{M}_{2,1}(z_1,z_2)}{\widetilde{M}_{2,1}(b_2(z_1),\widetilde{K}_{2,1}(z_1))}.$$

REMARK 7.4.4. When one wishes to relax the exhaustive assumption or the two-queue assumption, the analysis becomes intrinsically harder, as the model does not satisfy the branching property anymore. However, although an analysis in the spirit of Section 7.3 indeed seems hard to perform when dropping the exhaustive assumption, preliminary investigations suggest that the heavy-traffic limits of the waiting times and queue lengths still allow for compact and closed-form expressions. For instance, in the case of two-queue Markovian models with gated service (i.e. during a visit period, the server only serves the customers that were present at the start of this period), the heavy-traffic limits seem to have a similar connection with the heavy-traffic limits of a cyclic polling model as the one established in this chapter for the exhaustive case. The service discipline of this cyclic model, however, amounts to the κ-gated discipline as introduced in [260], but where κ is a geometric random variable rather than a constant. As this 'geometric gated' service discipline defies the branching property as well, heavy-traffic asymptotics for the cyclic equivalent are not readily available in the literature. As for the two-queue assumption, although an equivalent of Theorem 7.3.3 seems hard to find for this case, functional equations similar to (7.2) and (7.3) exist for a larger number of queues (cf. (8.7)). A heavy-traffic analysis may be found by carefully inspecting the behaviour of this functional equation under heavy-traffic scalings. In the next chapter, we drop both assumptions simultaneously and derive (cross-)moments of the joint distribution of the queue lengths.

Appendix

7.A Proof of Lemma 7.3.1

PROOF. We first focus on the value of $\left|1-b_1^{(\infty)}(z_2)\right|=\lim_{j\to\infty}\left|1-b_1^{(j)}(z_2)\right|$. For arbitrary $j>0$, we have for any z_2 in the unit circle that

$$\begin{aligned}\left|1-b_1^{(j)}(z_2)\right| &= \left|1-b_1(b_1^{j-1}(z_2))\right| \\ &= \left|\int_{t=0}^{\infty}(1-e^{-\lambda_1(1-\widetilde{K}_{1,2}(b_1^{(j-1)}(z_2)))t})d\mathbb{P}(\Gamma_2<t)\right| \\ &\leq \int_{t=0}^{\infty}\left|1-e^{-\lambda_1(1-\widetilde{K}_{1,2}(b_1^{(j-1)}(z_2)))t}\right|d\mathbb{P}(\Gamma_2<t),\end{aligned}$$

where the inequality constitutes the triangle inequality. Note that $|1-e^{-x}| \leq |x|$ for any $x \in \{z : z \in \mathbb{C} \wedge \Re(z) > 0\}$, so that

$$\begin{aligned}\left|1-b_1^{(j)}(z_2)\right| &\leq \int_{t=0}^{\infty} \lambda_1 t \left|1-\widetilde{K}_{1,2}(b_1^{(j-1)}(z_2))\right| d\mathbb{P}(\Gamma_2 < t) \\ &= \lambda_1 \mathbb{E}[\Gamma_2] \left|1-\widetilde{K}_{1,2}(b_1^{(j-1)}(z_2))\right| \\ &\leq \lambda_1 \mathbb{E}[\Gamma_2] \left|\int_{t=0}^{\infty} (1-e^{-\lambda_2(1-b_1^{(j-1)}(z_2))t}) d\mathbb{P}(\Gamma_1 < t)\right| \\ &\leq \lambda_1 \mathbb{E}[\Gamma_2]\lambda_2 \mathbb{E}[\Gamma_1] \left|1-b_1^{(j-1)}(z_2)\right|. \end{aligned} \tag{7.32}$$

Iteration of (7.32) leads to

$$\left|1-b_1^{(j)}(z_2)\right| \leq (\lambda_1 \mathbb{E}[\Gamma_2]\lambda_2 \mathbb{E}[\Gamma_1])^j |1-z_2|. \tag{7.33}$$

By (7.1), we have that $\mathbb{E}[\Gamma_i] = \mathbb{E}[B_i](1-\rho_i)^{-1}$, so that

$$\lambda_1 \mathbb{E}[\Gamma_2]\lambda_2 \mathbb{E}[\Gamma_1] = \frac{\rho_1}{1-\rho_2}\frac{\rho_2}{1-\rho_1} < 1. \tag{7.34}$$

The inequality follows since the queues are assumed to be stable, i.e. $0 \leq \rho < 1$. Therefore, $\rho_1 = \rho - \rho_2 < 1-\rho_2$ and similarly $\rho_2 < 1-\rho_1$. A combination of (7.32) and (7.34) now leads to

$$0 \leq \lim_{j\to\infty} \left|1-b_1^{(j)}(z_2)\right| \leq \lim_{j\to\infty} (\lambda_1 \mathbb{E}[\Gamma_2]\lambda_2 \mathbb{E}[\Gamma_1])^j |1-z_2| = 0.$$

Since $\lim_{j\to\infty} \left|1-b_1^{(j)}(z_2)\right| = 0$, we must have that $b_1^{(\infty)}(z_2) = \lim_{j\to\infty} b_1^{(j)}(z_2) = 1$.

By similar arguments, it can be shown that $b_2^{(\infty)}(z_1) = 1$ for any z_1 in the unit circle. Finally, it is evident that $\widetilde{K}_{1,2}(1) = \widetilde{K}_{2,1}(1) = \widetilde{F}_1(1,1) = \widetilde{F}_2(1,1) = 1$. The lemma now follows. □

7.B Proof of Lemma 7.3.2

PROOF. We initially focus on the product $\prod_{j=0}^{\infty} a_1(b_1^{(j)}(z_2))$. By the theory of infinite products (see e.g. [239, Chapter 1]), we have that $\prod_{j=0}^{\infty} a_1(b_1^{(j)}(z_2))$ converges if and only if $\sum_{j=0}^{\infty}(1-a_1(b_1^{(j)}(z_2)))$ converges. To establish the latter, it is enough to prove that the series $\sum_{j=0}^{\infty}|1-a_1(b_1^{(j)}(z_2))|$ converges. We observe that

$$\begin{aligned}&\left|1-a_1(b_1^{(j)}(z_2))\right| \\ &= \left|1-\frac{\nu_{2,1}\widetilde{M}_{2,1}(\widetilde{K}_{1,2}(b_1^{(j)}(z_2)), b_1^{(j)}(z_2))}{1-\nu_{1,1}\widetilde{M}_{1,1}(\widetilde{K}_{1,2}(b_1^{(j)}(z_2)), b_1^{(j)}(z_2))}\frac{\nu_{1,2}\widetilde{M}_{1,2}(\widetilde{K}_{1,2}(b_1^{(j)}(z_2)), b_1^{(j)}(z_2))}{1-\nu_{2,2}\widetilde{M}_{2,2}(\widetilde{K}_{1,2}(b_1^{(j)}(z_2)), b_1^{(j)}(z_2))}\right| \\ &= \left|\frac{\sum_{i=1}^{2} A_{1,i}(b_1^{(j)}(z_2))(1-\widetilde{M}_{i,1}(\widetilde{K}_{1,2}(b_1^{(j)}(z_2)), b_1^{(j)}(z_2)))}{D(z_2)}\right.\end{aligned}$$

$$+ \frac{\sum_{i=1}^{2} A_{2,i}(b_1^{(j)}(z_2))(1-\widetilde{M}_{i,2}(\widetilde{K}_{1,2}(b_1^{(j)}(z_2)), b_1^{(j)}(z_2)))}{D(z_2)}\Bigg|, \tag{7.35}$$

where

$$\begin{aligned}
A_{1,1}(z_2) &= \nu_{1,1}(1-\nu_{2,2}),\\
A_{1,2}(z_2) &= (1-\nu_{1,1})(1-\nu_{2,2}),\\
A_{2,1}(z_2) &= (1-\nu_{1,1})(1-\nu_{2,2})\widetilde{M}_{1,2}(\widetilde{K}_{1,2}(z_2), z_2),\\
A_{2,2}(z_2) &= \nu_{2,2}(1-\nu_{1,1}\widetilde{M}_{1,1}(\widetilde{K}_{1,2}(z_2), z_2)) \text{ and}\\
D(z_2) &= (1-\nu_{1,1}\widetilde{M}_{1,1}(\widetilde{K}_{1,2}(b_1^{(j)}(z_2)), b_1^{(j)}(z_2)))(1-\nu_{2,2}\widetilde{M}_{2,2}(\widetilde{K}_{1,2}(b_1^{(j)}(z_2)), b_1^{(j)}(z_2))).
\end{aligned}$$

Using the triangle inequality and similar arguments as those in the proof of Lemma 7.3.1, we note that for $1 \le i,k \le 2$ and $j > 0$,

$$\begin{aligned}
&\left|1-\widetilde{M}_{i,k}(\widetilde{K}_{1,2}(b_1^{(j)}(z_2)), b_1^{(j)}(z_2))\right|\\
&\le \int_{t=0}^{\infty} \left|1-e^{-(\lambda_1(1-\widetilde{K}_{1,2}(b_1^{(j)}(z_2)))+\lambda_2(1-b_1^{(j)}(z_2)))t}\right| d\mathbb{P}(S_{i,k} < t)\\
&\le \mathbb{E}[S_{i,k}]\left(\lambda_1\left|1-\widetilde{K}_{1,2}(b_1^{(j)}(z_2))\right| + \lambda_2\left|1-b_1^{(j)}(z_2)\right|\right)\\
&\le \mathbb{E}[S_{i,k}]\lambda_2(\lambda_1\mathbb{E}[\Gamma_1]+1)\left|1-b_1^{(j)}(z_2)\right|.
\end{aligned}$$

Moreover, it is trivially seen that $\left|A_{i,k}(z_2)\right| \le 1$ for $1 \le i,k \le 2$ and any z_2 in the unit circle. Furthermore, since $\left|\widetilde{M}_{i,k}(\widetilde{K}_{1,2}(z_2), z_2)\right| \le 1$, we have that $|D(z_2)| \ge (1-\nu_{1,1})(1-\nu_{2,2})$. Therefore, a combination of (7.33) and (7.35) with the triangle inequality leads to

$$\begin{aligned}
\left|1-a_1(b_1^{(j)}(z_2))\right| \le{}& \frac{\mathbb{E}[S_{1,1}]+\mathbb{E}[S_{1,2}]+\mathbb{E}[S_{2,1}]+\mathbb{E}[S_{2,2}]}{(1-\nu_{1,1})(1-\nu_{2,2})}\\
&\times \lambda_2(\lambda_1\mathbb{E}[\Gamma_1]+1)(\lambda_1\mathbb{E}[\Gamma_2]\lambda_2\mathbb{E}[\Gamma_1])^j\,|1-z_2|\,.
\end{aligned}$$

This result obviously shows, in combination with (7.34), that $\sum_{j=0}^{\infty}|1-a_1(b_1^{(j)}(z_2))|$ is bounded from above by a converging geometric sum. As a result, $\sum_{j=0}^{\infty}|1-a_1(b_1^{(j)}(z_2))|$ converges, so that $\prod_{j=0}^{\infty} a_1(b_1^{(j)}(z_2))$ converges. Finally, the convergence of the product $\prod_{j=0}^{\infty} a_2(b_2^{(j)}(z_1))$ can be established similarly. □

8

MANY-QUEUE MODELS WITH BRANCHING-TYPE SERVICE DISCIPLINES

Now that the waiting times and the queue lengths of two-queue models with exhaustive service are completely characterised and their heavy-traffic behaviour is identified, we study the general class of systems with an arbitrary number of queues and branching-type service disciplines at each of the queues in this chapter. This general case is significantly harder to analyse. Although we study branching-type service disciplines, i.e. service disciplines that in principle allow for the branching property as described in Section 1.3.2 to hold, the Markovian routing mechanism breaks down the branching structure in case of non-exhaustive service or a number of queues that is larger than two. Therefore, the analysis of the general case is more complicated. Nevertheless, we derive a functional equation for the (probability generating function of the) joint queue length distribution observed at a point in time at which the server visits a certain queue. From this functional equation, expressions for the (cross-)moments of the queue lengths follow. We also derive a pseudo-conservation law for this generalised class of polling systems. We will use these results in Chapter 9 for optimisation purposes.

8.1 Introduction

We now drop the assumptions of two queues and exhaustive service at each of the queues, which we made in Chapter 7. Instead, we now analyse Markovian polling systems with an arbitrary number of queues. Rather than just the exhaustive service discipline, we study the complete class of so-called *branching-type service disciplines*, i.e. service disciplines that would allow the system to satisfy the branching property that we discussed in Section 1.3.2 in case the server were to visit the queues in a cyclic order (see also [208]). This is a wide class of service disciplines. Common examples of branching-type service disciplines are the exhaustive service discipline and the gated service discipline. Under the gated service discipline, the server will already initiate a switch-over period when he served all of the customers at the current queue that were present at the start of the current visit period. Thus, customers arriving during a visit period will at least have to wait until the next time the server visits their queue. The class of branching-type service disciplines also includes lesser known service disciplines, such as the binomial gated discipline

introduced in [157] and its exhaustive counterpart as defined in [46] known as the binomial exhaustive discipline. Under the binomial gated discipline, when the server finds n customers present at the start of a visit period at queue j, he will serve a binomial (n, r_j) number of these customers before switching, $0 < r_j \leq 1$. Under the binomial exhaustive discipline, the server not only serves the binomial (n, r_j) number of the customers present at the start of the visit period, but subsequently also the type-j customers arriving during the service of these customers ('the children'), the type-j customers arriving during the service of the children ('the grandchildren') and so on. As a result, the expected number of type-j customers that are left behind by the server at the end of the visit period equals $n(1-r_j)$. We will give special attention to these binomial service disciplines in Chapter 9.

The class of polling systems that we study in this chapter includes the class of polling models studied in Chapter 7, but is in fact much broader. This class is much harder to analyse. As already mentioned in Remark 7.4.4, the absence of the exhaustive and two-queue assumptions leads to the fact that the queue length vector cannot be modelled as a multi-type branching process with immigration. This considerably complicates the identification of the complete distributions of the joint queue length and other performance measures.

Despite the violation of the branching property, expressions for (cross-)moments of the joint queue length distribution can still be derived. In this chapter, we use the following strategy to achieve this goal. After introducing the necessary notation in Section 8.2, we use an approach similar to the *buffer occupancy* method introduced in [68, 69] to derive (cross-)moments of the joint queue length distribution at polling epochs in Section 8.3. More specifically, we first derive a functional equation for the probability generating function of the distribution of the joint queue length at polling epochs. By differentiation, this leads to the derivation of the (cross-)moments of this distribution. We extend this analysis in Section 8.4 to allow for the computation of (cross-)moments of the joint queue length at an arbitrary point in time. Finally, as a by-product, we also obtain an explicit expression for the expected amount of waiting work in the system in Section 8.5 based on the concept of the so-called pseudo-conservation law obtained in [49].

8.2 Notation

Much of the notation that we will use in this chapter to study the Markovian polling model under general assumptions is similar to the notation used in Chapter 7. Nevertheless, to accommodate the broader model assumptions, we give below an exhaustive overview of the notation used in this chapter.

The model now consists of $N \geq 2$ infinite-buffer queues, $Q_1, \ldots, Q_N$, and a single server. Customers arriving at Q_i, also referred to as type-i customers, do so according to a Poisson process with intensity λ_i. The generic service requirement of a type-i customer is represented by the random variable B_i, of which the Laplace-Stieltjes transform is given by $\widetilde{B}_i(s) = \mathbb{E}[e^{-sB_i}]$. The load that Q_i brings to the system is denoted by $\rho_i = \lambda_i \mathbb{E}[B_i]$. We assume throughout this chapter that the aggregate load $\rho = \sum_{i=1}^{N} \rho_i$ is less than one.

All the queues share a single server. However, this server can only serve customers of one queue at a time. Hence, after serving a given number of customers at one queue (a visit period), the server will switch over to another queue to start service there. We assume that the server adheres to a Markovian routing scheme. Thus, the position of

the server is governed by an irreducible discrete-time Markov chain $\{Z_m, m \geq 0\}$ on the state space $\mathscr{S} = \{1, \ldots, N\}$. As a result, the queue being served during the m-th visit period is Q_{Z_m}. The one-step transition probability matrix corresponding to the discrete-time Markov chain $\{Z_m, m \geq 0\}$ is given by $\boldsymbol{P} = (p_{i,j})_{i,j \in \mathscr{S}}$, and its unique invariant probability measure denoted by $\boldsymbol{q} = (q_i)_{i \in \mathscr{S}}$ satisfies the conditions $\boldsymbol{qP} = \boldsymbol{q}$ and $\sum_{j=1}^{N} q_j = 1$. In short, after completing a visit period to Q_i, the server will switch over to Q_j with probability $p_{i,j}$. Such a setup from Q_i to Q_j takes a continuously distributed random amount of time $S_{i,j}$ (also referred to as the switch-over time), of which the Laplace-Stieltjes transform is given by $\widetilde{S}_{i,j}(s) = \mathbb{E}[e^{-sS_{i,j}}]$. We assume all interarrival times, service times and switch-over times in the model to be independent.

The number of customers that are served during a visit period V_i at Q_i is governed by the service discipline at Q_i. We do not limit our analysis to a single service discipline, but we assume that the service discipline at each of the queues belongs to the class of service disciplines that satisfy the following property.

PROPERTY 8.2.1. If the server arrives at Q_i to find l_i customers there, then during the course of the server's visit, each of these l_i customers will effectively be replaced in an independent and identically distributed manner by a random population having a probability generating function $\widetilde{H}_i(\boldsymbol{z}) = \widetilde{H}_i(z_1, \ldots, z_N)$, which is called the *offspring function* and can be any N-dimensional probability generating function.

We particularly consider this class of service disciplines since it covers a wide range of commonly adapted policies. At the same time, it still allows for a tractable analysis. Observe that cyclic polling systems where the service disciplines satisfy this property allow their joint queue length processes to be modelled as multi-type branching processes with immigration. This is not necessarily the case for Markovian polling systems; as we already noted before, the queue lengths in a Markovian polling system generally do not allow for such an interpretation.

Two service disciplines satisfying this property that will receive specific attention in the next chapter are the binomial gated and the binomial exhaustive service discipline. Under the binomial gated discipline, the number of type-i customers that are served during a visit period, at the start of which m_i type-i customers are present in the system, is binomially distributed with parameters m_i and r_i, $r_i \in (0, 1]$. Thus, a type-i customer present at the start of a non-empty visit period is still present at the end of this period with probability $1-r_i$ or is served during this period with probability r_i. Since new customers will arrive at each of the queues during the service of a type-i customer, the offspring function is in this case given by $\widetilde{H}_i(\boldsymbol{z}) = (1-r_i)z_i + r_i \widetilde{B}_i(\sum_{j \in \mathscr{S}} \lambda_j (1-z_j))$. The binomial exhaustive discipline has many similarities with the binomial gated discipline. Again, a type-i customer present at the start of a visit period remains in the system with probability $1 - r_i$. However, with probability r_i, not only the customer itself will be served during the visit period, but also all of its type-i offspring (thus, the type-i 'children' that arrive during this service time, the type-i 'grandchildren' that arrive during the service times of the children and so on). Therefore, the visit period now consists of a number of type-i busy periods (i.e. periods of time needed to serve a type-i customer and all of its type-i offspring) that is binomially distributed with parameters m_i and r_i. When denoting the duration of such a busy period generated by a type-i customer by Γ_i, and its corresponding Laplace-Stieltjes transform by $\widetilde{\Gamma}_i(s) = \mathbb{E}[e^{-s\Gamma_i}]$, the offspring function of a queue adhering to the binomial exhaustive service discipline is thus given by $\widetilde{H}_i(\boldsymbol{z}) = (1 - r_i)z_i + r_i \widetilde{\Gamma}_i(\sum_{j \in \mathscr{S} \setminus \{i\}} \lambda_j (1 - z_j))$. In both

of these service disciplines, r_i is a measure of the service exhaustiveness; the higher r_i, the more customers the server will serve on average at Q_i over the course of a non-empty visit period. Therefore, we also refer to r_i as the exhaustiveness probability. Observe that for $r_i = 1$, the binomial gated and binomial exhaustive service disciplines as described above reduce to the classical gated and exhaustive service disciplines.

We denote by C_i the time between two consecutive points in time at which the server *polls* Q_i (also called polling epochs or polling instants). A server is said to poll a queue when he starts a visit period at that queue. The time C_i consists of an average of $1/q_i$ visit periods and subsequent switch-over periods by virtue of the Markovian routing dynamics. Furthermore, any arbitrary visit period and subsequent switch-over period in stationarity corresponds to a visit to Q_i with probability q_i. Thus, there are on average q_j/q_i visit periods and setups to Q_j between two polling epochs of Q_i. The expected time the server takes for a visit to Q_j and the subsequent setup equals $\mathbb{E}[V_j] + \sum_{k\in\mathscr{S}} p_{j,k}\mathbb{E}[S_{j,k}]$. As a consequence,

$$\mathbb{E}[C_i] = \frac{1}{q_i}\sum_{j\in\mathscr{S}} q_j(\mathbb{E}[V_j] + \sum_{k\in\mathscr{S}} p_{j,k}\mathbb{E}[S_{j,k}]). \tag{8.1}$$

It follows from balance arguments that $\mathbb{E}[V_i] = \rho_i\mathbb{E}[C_i]$ and $\frac{q_i\mathbb{E}[V_i]}{q_j\mathbb{E}[V_j]} = \frac{\rho_i}{\rho_j}$. As a result, we have by (8.1) that, for every $i \in \mathscr{S}$,

$$\mathbb{E}[C_i] = \frac{\sigma}{q_i(1-\rho)}, \tag{8.2}$$

where $\sigma = \sum_{j\in\mathscr{S}} q_j \sum_{k\in\mathscr{S}} p_{j,k}\mathbb{E}[S_{j,k}]$ (see also [54]). Note that σ represents the overall mean of the switch-over times incurred by the server. We denote by ζ_i the reciprocal of the expected number of customers served by the server during a visit period V_i. We thus have that $\zeta_i = \frac{\mathbb{E}[B_i]}{\mathbb{E}[V_i]} = \frac{1}{\lambda_i\mathbb{E}[C_i]}$.

In the remainder of this chapter, we are interested in the joint queue length distributions (including any customer in service) at several time epochs. To this end, we denote by $\boldsymbol{F}_i = (F_{i,1},\ldots,F_{i,N})$ the joint stationary queue length conditioned on the event that the server currently polls Q_i. The vectors $\boldsymbol{G}_i$, $\boldsymbol{M}_i$, $\boldsymbol{N}_i$, $\boldsymbol{X}_i$ and $\boldsymbol{Y}_{i,j}$ similarly represent the joint stationary queue length observed at a point in time at which the server ends a visit period at Q_i, the server starts serving a type-i customer, the server completes service of a type-i customer, the server is serving customers at Q_i and the server is currently switching from Q_i to Q_j, respectively. The unconditional stationary joint queue length of the queues in the system is given by $\boldsymbol{L}$. For an arbitrary N-dimensional random variable $\boldsymbol{R} = (R_1,\ldots,R_N)$, we denote its N-dimensional probability generating function by $\widetilde{R}(\boldsymbol{z}) = \widetilde{R}(z_1,\ldots,z_N) = \mathbb{E}[\prod_{k\in\mathscr{S}} z_k^{R_k}]$. Furthermore, we define $\widetilde{R}^{(k)}(\boldsymbol{z}) = \frac{\partial}{\partial z_k}\widetilde{R}(\boldsymbol{z})$, $\widetilde{R}^{(k,l)}(\boldsymbol{z}) = \frac{\partial}{\partial z_l}\frac{\partial}{\partial z_k}\widetilde{R}(\boldsymbol{z})$, $r(k) = \widetilde{R}^{(k)}(\boldsymbol{z})|_{\boldsymbol{z}=\boldsymbol{1}}$ and $r(k,l) = \widetilde{R}^{(k,l)}(\boldsymbol{z})|_{\boldsymbol{z}=\boldsymbol{1}}$. Thus, we use lower cases to refer to derivatives of probability generating functions evaluated at $\boldsymbol{z} = \boldsymbol{1}$. It holds that $r(k) = \mathbb{E}[R_k]$, $r(k,k) = \mathbb{E}[R_k^2] - \mathbb{E}[R_k]$ and $r(k,l) = \mathbb{E}[R_kR_l]$ if $k \neq l$. So, for example, $f_i(k)$ denotes the mean queue length of Q_k when the server polls Q_i. Likewise, $f_i(k,l)$ refers to the second-order cross-moment pertaining to the queue lengths of Q_k and Q_l when the server polls Q_i and $k \neq l$. Besides the shorthand notation $\boldsymbol{z} = (z_1,\ldots,z_N)$ that we used above, we will also use $\boldsymbol{z}_i^H = (z_1,\ldots,z_{i-1},H_i(\boldsymbol{z}),z_{i+1},\ldots,z_N)$ and $\Sigma(\boldsymbol{z}) = \sum_{k\in\mathscr{S}} \lambda_k(1-z_k)$. Observe that the distribution of the waiting time W_i for type-i customers with Laplace-Stieltjes transform $\widetilde{W}_i(s) = \mathbb{E}[e^{-sW_i}]$ is related to the queue length L_i through the distributional form of Little's law $\widetilde{W}_i(s) = \frac{\widetilde{L}_i(1-s/\lambda_i)}{\widetilde{B}_i(s)}$, as shown in [135].

8.3 Joint queue length at polling epochs

We now derive a functional equation for the probability generating function $\widetilde{F}_i(\boldsymbol{z})$ of the queue length distribution conditioned on the event that the server polls Q_i, $i \in \mathscr{S}$. Based on this functional equation, all moments of the joint queue length distribution at a polling epoch of Q_i can be derived. In particular, we show how to derive solvable sets of equations for $f_i(k)$ and $f_i(k,l)$, $k,l \in \mathscr{S}$. From these sets, we obtain expressions for the first and second-order (cross-)moments of the joint queue length distribution at polling epochs. We note that by using the same methodology, expressions for higher-order moments can be derived.

8.3.1 Functional equation

To obtain a functional equation for $\widetilde{F}_i(\boldsymbol{z})$, we first relate the joint queue length distribution at a polling epoch of Q_i to the queue length distribution at the preceding polling instant at any queue. To this end, recall that Z_m refers to the index of the queue that the server visits at the m-th polling instant. Furthermore, let $\boldsymbol{J}_m = (J_{m,1}, \ldots, J_{m,N})$ and $\boldsymbol{K}_m = (K_{m,1}, \ldots, K_{m,N})$ be the joint queue length at the start of the m-th visit period (to any queue) since the startup of the system and its end, respectively. By conditioning on Z_m and Z_{m+1}, we have that

$$\begin{aligned}&\mathbb{E}\left[\mathbb{1}_{\{Z_{m+1}=j\}}\prod_{k\in\mathscr{S}} z_k^{J_{m+1,k}}\right]\\ &=\sum_{i\in\mathscr{S}}\mathbb{P}(Z_{m+1}=j \mid Z_m=i)\mathbb{P}(Z_m=i)\mathbb{E}\left[\prod_{k\in\mathscr{S}} z_k^{J_{m+1,k}} \mid Z_{m+1}=j, Z_m=i\right].\end{aligned} \tag{8.3}$$

Observe that, as per Property 8.2.1, the total population in the system during the m-th visit period only changes through the replacement of every type-Z_m customer by a population with probability generating function $\widetilde{H}_{Z_m}(\boldsymbol{z})$. More colloquially speaking, the type-Z_m customers that get served during the m-th visit period allow new customers of any type to arrive to the system over the course of this visit period. As the number of arriving customers of any type is independent of $J_{m,i}, i \in \mathscr{S}\backslash\{Z_m\}$, we have that

$$\begin{aligned}\mathbb{E}\left[\prod_{k\in\mathscr{S}} z_k^{K_{m,k}} \mid Z_m=i\right] &= \mathbb{E}\left[\prod_{k\in\mathscr{S}\backslash\{i\}} z_k^{J_{m,k}} \mid Z_m=i\right]\sum_{n=0}^{\infty}(\widetilde{H}_i(\boldsymbol{z}))^n\mathbb{P}(J_{m,i}=n)\\ &= \mathbb{E}\left[(\widetilde{H}_i(\boldsymbol{z}))^{J_{m,i}}\prod_{k\in\mathscr{S}\backslash\{i\}} z_k^{J_{m,k}} \mid Z_m=i\right].\end{aligned} \tag{8.4}$$

Furthermore, the population at the start of the $(m+1)$-st visit period consists of the customers already there at the end of the m-th visit period and the customers that arrive during the subsequent switch-over period according to type-specific Poisson processes. As these two subpopulations are independent, we obtain

$$\mathbb{E}\left[\prod_{k\in\mathscr{S}} z_k^{J_{m+1,k}} \mid Z_{m+1}=j, Z_m=i\right]$$

$$= \mathbb{E}\left[\prod_{k\in\mathscr{S}} z_k^{K_{m,k}} \mid Z_m = i\right] \int_{t=0}^{\infty} \sum_{n_1=0}^{\infty} \cdots \sum_{n_N=0}^{\infty} \prod_{k\in\mathscr{S}} z_k^{n_k} e^{-\lambda_k t} \frac{(\lambda_k t)^{n_k}}{n_k!} d\mathbb{P}(S_{i,j} < t)$$

$$= \mathbb{E}\left[\prod_{k\in\mathscr{S}} z_k^{K_{m,k}} \mid Z_m = i\right] \widetilde{S}_{i,j}(\Sigma(z)). \tag{8.5}$$

Combining (8.3)–(8.5) with the fact that $\mathbb{P}(Z_{m+1} = j \mid Z_m = i) = p_{i,j}$ now gives

$$\mathbb{E}\left[\mathbb{1}_{\{Z_{m+1}=j\}} \prod_{k\in\mathscr{S}} z_k^{J_{m+1,k}}\right]$$

$$= \sum_{i\in\mathscr{S}} p_{i,j}\mathbb{P}(Z_m = i)\mathbb{E}\left[(\widetilde{H}_i(z))^{J_{m,i}} \prod_{k\in\mathscr{S}\setminus\{i\}} z_k^{J_{m,k}} \mid Z_m = i\right] \widetilde{S}_{i,j}(\Sigma(z)). \tag{8.6}$$

The left-hand side of (8.6) can be rewritten as

$$\mathbb{E}[\mathbb{1}_{\{Z_{m+1}=j\}} \prod_{k\in\mathscr{S}} z_k^{J_{m+1,k}}] = \mathbb{P}(Z_{m+1} = j)\mathbb{E}\left[\prod_{k\in\mathscr{S}} z_k^{J_{m+1,k}} \mid Z_{m+1} = j\right].$$

Furthermore, we have by definition that

$$\lim_{m\to\infty} \mathbb{P}(Z_m = i) = q_i \text{ and } \lim_{m\to\infty} \mathbb{E}\left[\prod_{k\in\mathscr{S}} z_k^{J_{m,k}} \mid Z_m = i\right] = \widetilde{F}_i(z).$$

Hence, by letting $m \to \infty$ and recalling that $z_i^H = (z_1, \ldots, z_{i-1}, H_i(z), z_{i+1}, \ldots, z_N)$, (8.6) implies the following functional equation for all $i, j \in \mathscr{S}$:

$$q_j \widetilde{F}_j(z) = \sum_{i\in\mathscr{S}} p_{i,j} q_i \widetilde{F}_i(z_i^H) \widetilde{S}_{i,j}(\Sigma(z)). \tag{8.7}$$

Observe that for $N = 2$ and $\widetilde{H}_i(z) = \widetilde{\Gamma}_i(\sum_{j\in\mathscr{S}\setminus\{i\}} \lambda_j(1-z_j))$ (i.e. exhaustive service), (8.7) reduces to (7.2) and (7.3), the functional equations found for the two-queue exhaustive model.

8.3.2 Queue length moments at polling epochs

From the functional equation (8.7), an explicit expression for $\widetilde{F}_i(z)$ is not easily derived. However, using this functional equation, all (cross-)moments of the queue lengths can be computed. We show how to compute the first-order and second-order (cross-)moments of the marginal queue lengths found in the system at polling instants. Higher-order (cross-)moments can be computed through the same methodology at the cost of a larger computational complexity.

First, recall that (cross-)moments of the (conditional) queue length vector are given by $\mathbb{E}[L_l \mid \text{server just polled } Q_j] = \mathbb{E}[F_{j,l}] = f_j(l)$, $\mathbb{E}[L_l^2 \mid \text{server just polled } Q_j] = \mathbb{E}[F_{j,l}^2] = f_j(l) + f_j(l,l)$ and that $\mathbb{E}[L_l L_m \mid \text{server just polled } Q_j] = \mathbb{E}[F_{j,l}F_{j,m}] = f_j(l,m)$ for any

$j,l,m \in \mathscr{S}$, $l \neq m$. To obtain these numbers, we first take the derivative with respect to z_l in both sides of (8.7). For $j,l \in \mathscr{S}$, this leads to

$$q_j \widetilde{F}_j^{(l)}(z) = \sum_{i\in\mathscr{S}} p_{i,j} q_i \lambda_l \mathbb{E}[S_{i,j} e^{-(\Sigma(z))S_{i,j}}] \widetilde{F}_i(z_i^H) + \sum_{i\in\mathscr{S}\setminus\{l\}} p_{i,j} q_i \tilde{S}_{i,j}(\Sigma(z)) \widetilde{F}_i^{(l)}(z_i^H) + \sum_{i\in\mathscr{S}} p_{i,j} q_i \widetilde{S}_{i,j}(\Sigma(z)) \widetilde{H}_i^{(l)}(z) \widetilde{F}_i^{(i)}(z_i^H). \tag{8.8}$$

Evaluating this equation in the point $z = \mathbf{1}$ subsequently leads to

$$q_j f_j(l) = \sum_{i\in\mathscr{S}} p_{i,j} q_i \lambda_l \mathbb{E}[S_{i,j}] + \sum_{i\in\mathscr{S}\setminus\{l\}} p_{i,j} q_i f_i(l) + \sum_{i\in\mathscr{S}} p_{i,j} q_i h_i(l) f_i(i). \tag{8.9}$$

This set of N^2 equations leads to expressions for $f_j(l)$, $j,l \in \mathscr{S}$. If one is only interested in the values of $f_j(l)$ for a specific value of l, the complexity of these computations can be reduced to only solving a set of N equations, since an explicit expression for $f_i(i)$ is available; see Remark 8.3.1.

To find a similar set of equations for $f_j(l,m)$, $j,l,m \in \mathscr{S}$, we first derive a functional equation for $\widetilde{F}_j^{(l,m)}(z)$. By differentiating both sides of (8.8) with respect to z_m, $m \in \mathscr{S}$, we obtain

$$\begin{aligned}
&q_j \widetilde{F}_j^{(l,m)}(z) \\
&= \sum_{i\in\mathscr{S}} p_{i,j} q_i \lambda_l \lambda_m \mathbb{E}[S_{i,j}^2 e^{-\Sigma(z)S_{i,j}}] \widetilde{F}_i(z_i^H) + \sum_{i\in\mathscr{S}\setminus\{m\}} p_{i,j} q_i \lambda_l \mathbb{E}[S_{i,j} e^{-\Sigma(z)S_{i,j}}] \widetilde{F}_i^{(m)}(z_i^H) \\
&\quad + \sum_{i\in\mathscr{S}} p_{i,j} q_i \lambda_l \mathbb{E}[S_{i,j} e^{-\Sigma(z)S_{i,j}}] \widetilde{H}_i^{(m)}(z) \widetilde{F}_i^{(i)}(z_i^H) + \sum_{i\in\mathscr{S}\setminus\{l\}} p_{i,j} q_i \lambda_m \mathbb{E}[S_{i,j} e^{-\Sigma(z)S_{i,j}}] \widetilde{F}_i^{(l)}(z_i^H) \\
&\quad + \sum_{i\in\mathscr{S}\setminus\{l,m\}} p_{i,j} q_i \widetilde{S}_{i,j}(\Sigma(z)) \widetilde{F}_i^{(l,m)}(z_i^H) + \sum_{i\in\mathscr{S}\setminus\{l\}} p_{i,j} q_i \widetilde{S}_{i,j}(\Sigma(z)) \widetilde{H}_i^{(m)}(z) \widetilde{F}_i^{(i,l)}(z_i^H) \\
&\quad + \sum_{i\in\mathscr{S}} p_{i,j} q_i \lambda_m \mathbb{E}[S_{i,j} e^{-\Sigma(z)S_{i,j}}] \widetilde{H}_i^{(l)}(z) \widetilde{F}_i^{(i)}(z_i^H) + \sum_{i\in\mathscr{S}} p_{i,j} q_i \widetilde{S}_{i,j}(\Sigma(z)) \widetilde{H}_i^{(l,m)}(z) \widetilde{F}_i^{(i)}(z_i^H) \\
&\quad + \sum_{i\in\mathscr{S}} p_{i,j} q_i \widetilde{S}_{i,j}(\Sigma(z)) \widetilde{H}_i^{(l)}(z) \widetilde{H}_i^{(m)}(z) \widetilde{F}_i^{(i,i)}(z_i^H) \\
&\quad + \sum_{i\in\mathscr{S}\setminus\{m\}} p_{i,j} q_i \widetilde{S}_{i,j}(\Sigma(z)) \widetilde{H}_i^{(l)}(z) \widetilde{F}_i^{(i,m)}(z_i^H).
\end{aligned}$$

Similarly to the computations above, evaluating this equation in the point $z = \mathbf{1}$ leads to

$$\begin{aligned}
q_j f_j(l,m) &= \sum_{i\in\mathscr{S}} p_{i,j} q_i \lambda_l \lambda_m \mathbb{E}[S_{i,j}^2] + \sum_{i\in\mathscr{S}\setminus\{m\}} p_{i,j} q_i \lambda_l \mathbb{E}[S_{i,j}] f_i(m) \\
&\quad + \sum_{i\in\mathscr{S}} p_{i,j} q_i \lambda_l \mathbb{E}[S_{i,j}] h_i(m) f_i(i) + \sum_{i\in\mathscr{S}\setminus\{l\}} p_{i,j} q_i \lambda_m \mathbb{E}[S_{i,j}] f_i(l) \\
&\quad + \sum_{i\in\mathscr{S}\setminus\{l,m\}} p_{i,j} q_i f_i(l,m) + \sum_{i\in\mathscr{S}\setminus\{l\}} p_{i,j} q_i h_i(m) f_i(i,l) \\
&\quad + \sum_{i\in\mathscr{S}} p_{i,j} q_i \lambda_m \mathbb{E}[S_{i,j}] h_i(l) f_i(i) + \sum_{i\in\mathscr{S}} p_{i,j} q_i h_i(l,m) f_i(i)
\end{aligned}$$

$$+\sum_{i\in\mathscr{S}} p_{i,j}q_i h_i(l)h_i(m)f_i(i,i)+\sum_{i\in\mathscr{S}\setminus\{m\}} p_{i,j}q_i h_i(l)f_i(i,m). \tag{8.10}$$

For any $j,l,m\in\mathscr{S}$, this constitutes a set of N^3 equations for $f_j(l,m)$. Since expressions for $f_i(j)$, $i,j\in\mathscr{S}$, are known after solving the set of equations given in (8.9), expressions for $f_j(l,m)$ can now be calculated for all $j,l,m\in\mathscr{S}$. As mentioned above, the expressions for $f_j(l)$ and $f_j(l,m)$ then subsequently lead to expressions for the first-order and second-order (cross-)moments of the queue lengths when the server polls Q_j. Although these moments may be of separate interest, we also use the derived expressions for $f_j(l)$ and $f_j(l,m)$ in Section 8.4 to obtain moments for L_j, the queue length of Q_j at an arbitrary point in time. We finish this section with the observation that the sets of equations expressed in (8.9) and (8.10) are uniquely solvable, provided that the aggregate load ρ is smaller than one. One can confirm this by reducing (8.9) and (8.10) to equation sets of the form $A\boldsymbol{x}=\boldsymbol{b}$ and showing in a tedious, but straightforward way that the coefficient matrix A is invertible in that case.

REMARK 8.3.1. For any $i\in\mathscr{S}$, the term $f_i(i)$ can be computed explicitly. To do this, we make use of an observation most notably made by [87], which says that each time a visit beginning or a service completion occurs, this coincides with either a service beginning or a visit completion. All service beginning epochs in a visit period to Q_i are also service completion epochs at Q_i, except for the first service beginning epoch, because it is actually a visit beginning epoch. Likewise, all service completion epochs at Q_i are also service beginning epochs at that queue, except for the last service completion epoch, because it is actually a visit completion epoch. Since ζ_i denotes the fraction of service beginning (completion) epochs that also count as a visit beginning (completion) epoch, this observation leads to

$$\zeta_i\widetilde{F}_i(z)+\widetilde{N}_i(z)=\zeta_i\widetilde{G}_i(z)+\widetilde{M}_i(z), \tag{8.11}$$

or more specifically for the means,

$$\zeta_i f_i(i)+n_i(i)=\zeta_i g_i(i)+m_i(i). \tag{8.12}$$

Over the course of a service time at Q_i, on average ρ_i type-i customers arrive, after which one type-i customer leaves the system, because its service is completed. Therefore, $m_i(i)-n_i(i)=1-\rho_i$. Furthermore, we have by Property 8.2.1 that $g_i(i)=h_i(i)f_i(i)$. Relation (8.12) therefore reduces to an explicit expression for $f_i(i)$:

$$f_i(i)=\frac{1-\rho_i}{\zeta_i(1-h_i(i))}=\frac{\lambda_i\sigma(1-\rho_i)}{q_i(1-\rho)(1-h_i(i))}, \tag{8.13}$$

where the second equality follows from the fact that $\zeta_i=1/(\lambda_i\mathbb{E}[C_i])$ combined with (8.2).

8.4 Joint queue length at an arbitrary point in time

In Section 8.3, we have studied the probability generating function and moments of the joint queue length distribution at polling epochs. We now extend these results to obtain results for the queue lengths *at an arbitrary point in time*. We largely follow the approach of [51, Theorem 1] to express $\widetilde{L}(z)$, the probability generating function representing the

stationary joint queue length at an arbitrary point in time, in the conditional queue length probability generating functions studied above. Expressions for all (cross-)moments of the unconditional queue lengths in the moments of the queue lengths found at polling epochs subsequently follow from this relation.

To relate the unconditional queue length to the various conditional queue lengths studied before, we first observe that the server serves Q_i a fraction ρ_i of the time. At an arbitrary epoch during the remaining fraction $(1-\rho_i)$ of time, the server is in a switch-over process, which with probability $\frac{q_i p_{i,k} \mathbb{E}[S_{i,k}]}{\sigma}$ happens to be a setup from Q_i to Q_k. As a result, the unconditional probability generating function $\widetilde{L}(z)$ satisfies

$$\widetilde{L}(z) = \sum_{i \in \mathscr{S}} \left(\rho_i \widetilde{X}_i(z) + \frac{(1-\rho) q_i}{\sigma} \sum_{k \in \mathscr{S}} p_{i,k} \mathbb{E}[S_{i,k}] \widetilde{Y}_{i,k}(z) \right), \tag{8.14}$$

where the probability generating functions $\widetilde{X}_i(z)$ and $\widetilde{Y}_{i,k}(z)$ represent the joint queue lengths at arbitrary points during a visit period at Q_i and a switch-over period from Q_i to Q_k, respectively. The customer population present in the system at an arbitrary point in a visit period to Q_i is comprised of the population already there at the start of the current type-i service and the customers that have arrived during the past part of the current service period. As these two components are independent, we have that $\widetilde{X}_i(z) = \widetilde{M}_i(z) \frac{1-\widetilde{B}_i(\Sigma(z))}{\Sigma(z)\mathbb{E}[B_i]}$. Furthermore, it is easy to see that $\widetilde{N}_i(z) = z_i^{-1} \widetilde{B}_i(\Sigma(z)) \widetilde{M}_i(z)$. Combining these two relations with (8.11) leads to

$$\widetilde{X}_i(z) = \frac{\zeta_i}{\mathbb{E}[B_i]} \frac{z_i(\widetilde{F}_i(z) - \widetilde{G}_i(z))}{z_i - \widetilde{B}_i(\Sigma(z))} \frac{1-\widetilde{B}_i(\Sigma(z))}{\Sigma(z)}, \tag{8.15}$$

where, due to the fact that the service disciplines satisfy Property 8.2.1,

$$\widetilde{G}_i(z) = \widetilde{F}_i(z_i^H). \tag{8.16}$$

Similarly, as the customer population at an arbitrary point in a switch-over period from Q_i to Q_k is comprised of the population at the end of the past visit period to Q_i and the subsequent customer arrivals in the past part of the switch-over time, we have that

$$\widetilde{Y}_{i,k}(z) = \widetilde{G}_i(z) \frac{1-\widetilde{S}_{i,k}(\Sigma(z))}{\Sigma(z)\mathbb{E}[S_{i,k}]}. \tag{8.17}$$

A combination of the equations (8.14)–(8.17) leads to the unconditional probability generating function $\widetilde{L}(z)$ of the joint queue length expressed in the probability generating functions $\widetilde{F}_i(z)$ that represent the joint queue length at the moment the server polls Q_i.

We now show how one can use this relation to derive expressions for the unconditional mean marginal queue lengths $\mathbb{E}[L_i] = l(i)$. The same method can be used to obtain expressions for higher (cross-)moments, although the computations become lengthier. By using (8.15), we first obtain the first moment of $X_{i,i}$ as follows:

$$\begin{aligned} x_i(i) &= \lim_{z_i \uparrow 1} \frac{d}{dz_i} (\widetilde{X}_i(z)|_{z_k=1 \ \forall k \neq i}) \\ &= \lim_{z_i \uparrow 1} \frac{d}{dz_i} \left(\frac{\zeta_i}{\mathbb{E}[B_i]} \frac{z_i(\mathbb{E}[z_i^{F_{i,i}}] - \mathbb{E}[z_i^{G_{i,i}}])}{z_i - \widetilde{B}_i(\lambda_i(1-z_i))} \frac{1-\widetilde{B}_i(\lambda_i(1-z_i))}{\lambda_i(1-z_i)} \right) \end{aligned}$$

$$= \frac{\zeta_i(f_i(i)-g_i(i))}{1-\rho_i} + \frac{\lambda_i^2\zeta_i\mathbb{E}[B_i^2](f_i(i)-g_i(i))}{2(1-\rho_i)^2} + \frac{\zeta_i(f_i(i,i)-g_i(i,i))}{2(1-\rho_i)} + \frac{\lambda_i\mathbb{E}[B_i^2]}{2\mathbb{E}[B_i]}$$
$$= 1 + \frac{\lambda_i^2\mathbb{E}[B_i^2] + \zeta_i(f_i(i,i)(1-h_i(i)^2) - f_i(i)h_i(i,i))}{2(1-\rho_i)} + \frac{\lambda_i\mathbb{E}[B_i^2]}{2\mathbb{E}[B_i]}, \tag{8.18}$$

where we used in the fourth equality that $\zeta_i(f_i(i)-g_i(i)) = 1-\rho_i$ (cf. Remark 8.3.1) and the fact that

$$g_i(i,i) = f_i(i,i)(h_i(i))^2 + f_i(i)h_i(i,i)$$

due to (8.16). A similar but slightly shorter computation yields that the first moment of $X_{i,j}$, $i \neq j$, is given by

$$x_i(j) = \lim_{z_j\uparrow 1}\frac{d}{dz_j}(\widetilde{X}_i(z)|_{z_k=1\ \forall k\neq j}) = \lim_{z_j\uparrow 1}\frac{d}{dz_j}\left(\frac{\zeta_i}{\mathbb{E}[B_i]}\frac{\mathbb{E}[z_j^{F_{i,j}}]-\mathbb{E}[z_j^{G_{i,j}}]}{\lambda_j(1-z_j)}\right)$$
$$= \frac{\zeta_i(g_i(j,j)-f_i(j,j))}{2\lambda_j\mathbb{E}[B_i]}$$
$$= \frac{\zeta_i(2f_i(i,j)h_i(j) + f_i(i,i)(h_i(j))^2 + f_i(i)h_i(j,j))}{2\lambda_j\mathbb{E}[B_i]}, \tag{8.19}$$

where the fourth equality again follows from (8.16), which implies that

$$g_i(j,j) = f_i(j,j) + 2f_i(i,j)h_i(j) + f_i(i,i)(h_i(j))^2 + f_i(i)h_i(j,j).$$

As for the mean queue length $y_{i,k}(j)$ during a switch-over period from Q_i to Q_k, we have by (8.17) that, for all $i,j,k \in \mathscr{S}$,

$$y_{i,k}(j) = \lim_{z_j\uparrow 1}\frac{d}{dz_j}(\widetilde{Y}_i(z)|_{z_l=1\ \forall l\neq j}) = \lim_{z_j\uparrow 1}\frac{d}{dz_j}\left(\mathbb{E}[z_j^{G_{i,j}}]\frac{1-\widetilde{S}_{i,k}(\lambda_j(1-z_j))}{\lambda_j(1-z_j)\mathbb{E}[S_{i,k}]}\right)$$
$$= g_i(j) + \lambda_j\frac{\mathbb{E}[S_{i,k}^2]}{2\mathbb{E}[S_{i,k}]}$$
$$= \mathbb{1}_{\{j\neq i\}}f_i(j) + f_i(i)h_i(j) + \lambda_j\frac{\mathbb{E}[S_{i,k}^2]}{2\mathbb{E}[S_{i,k}]}, \tag{8.20}$$

where the last equality follows from

$$g_i(j) = \mathbb{1}_{\{j\neq i\}}f_i(j) + f_i(i)h_i(j),$$

which can be derived from (8.16).

We can now derive an expression for the unconditional mean queue length $\mathbb{E}[L_j]$ in terms of the f terms computed in the previous section. After differentiating both sides of (8.14) and evaluating the result in $z = \mathbf{1}$, we obtain

$$\mathbb{E}[L_j] = \sum_{i\in\mathscr{S}}\Big(\rho_i x_i(j) + \frac{(1-\rho)q_i}{\sigma}\sum_{k\in\mathscr{S}}p_{i,k}\mathbb{E}[S_{i,k}]y_{i,k}(j)\Big). \tag{8.21}$$

Since we already found expressions for $x_i(i)$, $x_i(j)$ and $y_{i,k}(j)$ in (8.18), (8.19) and (8.20), respectively, $\mathbb{E}[L_j]$ is now obtained in terms of moments of the queue lengths at polling instants, which we have already considered in Section 8.3.2. Note that from this expression

for the mean queue length, it is straightforward to derive expressions for the mean waiting time or the mean amount of waiting work present in the queue. In the next section, we provide an expression for the expected total amount of waiting work in the system.

8.5 Pseudo-conservation law

For polling systems with a server that visits the queues in a cyclic fashion, a stochastic decomposition for the stationary amount of work present in the system has been derived in [49]. In particular, the amount of work in the polling system can be decomposed into two independent terms: the amount of work in a corresponding M/G/1 system and the amount of work in the polling system at an arbitrary point in time during a switch-over period of the server. This decomposition allows for the derivation of a strikingly simple expression for the weighted sum $\sum_{i\in\mathcal{S}} \rho_i \mathbb{E}[W_i]$ of mean waiting times. This result is known as the pseudo-conservation law. Following [49], the pseudo-conservation law has been extended to allow for polling systems with Markovian routing in [54], but this extension only allows the server to serve the queues in an exhaustive, gated or one-limited manner exclusively. In this section, we further extend the pseudo-conservation law to allow for any branching-type service discipline.

In particular, it is shown in [54] that the expected amount of waiting work in polling systems with Markovian routing is given by

$$\sum_{i\in\mathcal{S}} \rho_i \mathbb{E}[W_i] = \rho \frac{\sum_{i\in\mathcal{S}} \lambda_i \mathbb{E}[B_i^2]}{2(1-\rho)} + \frac{1}{\sigma} \sum_{i\in\mathcal{S}} q_i \sum_{k\in\mathcal{S}} p_{i,k} \mathbb{E}[S_{i,k}] \mathbb{E}[\Psi_{i,k}], \tag{8.22}$$

where the latter term represents the expected amount of work in the system during a switch-over period and where $\mathbb{E}[\Psi_{i,k}]$ is the expected amount of work in the system when the server is in the process of switching from Q_i to Q_k. The authors in [54] then determine $\mathbb{E}[\Psi_{i,k}]$ for the exhaustive, gated and one-limited service discipline. Observe, however, that the expected amount of work in the system equals the sum of the (remaining) service requirements of all the customers present in the system. As a result, we have for a switch-over period from Q_i to Q_k that

$$\mathbb{E}[\Psi_{i,k}] = \sum_{j\in\mathcal{S}} y_{i,k}(j) \mathbb{E}[B_j]. \tag{8.23}$$

By combining (8.22) and (8.23) with (8.20) and (8.13), respectively, we thus have for the general case that

$$\begin{aligned}\sum_{i\in\mathcal{S}} \rho_i \mathbb{E}[W_i] = \rho \frac{\sum_{i\in\mathcal{S}} \lambda_i \mathbb{E}[B_i^2]}{2(1-\rho)} &+ \frac{1}{\sigma} \sum_{i\in\mathcal{S}} q_i \sum_{k\in\mathcal{S}} p_{i,k} \mathbb{E}[S_{i,k}] \sum_{j\in\mathcal{S}\setminus\{i\}} f_i(j) \mathbb{E}[B_j] \\ &+ \frac{1}{1-\rho} \sum_{i\in\mathcal{S}} \lambda_i \frac{1-\rho_i}{1-h_i(i)} \sum_{k\in\mathcal{S}} p_{i,k} \mathbb{E}[S_{i,k}] \sum_{j\in\mathcal{S}} \mathbb{E}[B_j] h_i(j) \\ &+ \frac{\rho}{2\sigma} \sum_{i\in\mathcal{S}} q_i \sum_{k\in\mathcal{S}} p_{i,k} \mathbb{E}[S_{i,k}^2],\end{aligned} \tag{8.24}$$

provided that the service discipline pertaining to each queue satisfies Property 8.2.1. This expression uses the $f_i(j)$ terms that we computed in Section 8.3.2. In the next chapter, we will use this newly derived pseudo-conservation law for optimisation purposes.

9

OPTIMISATION WITH AN APPLICATION TO WIRELESS RANDOM-ACCESS NETWORKS

In Chapters 7 and 8, we analysed performance measures of the Markovian polling model under various assumptions. While the results obtained are of independent interest, we focus in this chapter on their application to wireless random-access networks. In particular, we address several optimisation questions of how to choose certain model parameters so as to minimise a (weighted) sum of mean queue lengths. Implementation of the resulting solutions in wireless random-access networks, however, in principle requires each node to have complete information on each of the other nodes present in the network. This is not a valid assumption in practice due to the decentralised nature of these networks. Therefore, we also present an adaptive control algorithm for finding the optimal parameter values in a distributed fashion by having the nodes use measurements of the time between two subsequent periods of activity in the medium.

9.1 Introduction

In this chapter, we focus on the application of Markovian polling systems to wireless random-access networks. As already explained in Section 1.3.2, the various queues in the polling system correspond to packet buffers at several wireless transmitters (or, nodes), which need to share the medium in a mutually exclusive way because of interference. In wireless random-access networks, carrier-sense multiple-access collision-avoidance algorithms are usually implemented, which provide a common mechanism for governing the use of the medium by the transmitters in a distributed fashion. In these algorithms, the nodes obey random back-off times between periods of activity. This is done not only to avoid collisions, but also to give the other nodes an opportunity to become active.

We assume that the nodes implement back-off times that are independent and exponentially distributed with a rate ν_i, which we refer to as the back-off rate. The relative values of the back-off rates indicate the relative priority of transmission among the N nodes. In other words, a low-priority node aims to be in back-off much longer than a high-priority node and thus adheres to a smaller back-off rate. Because of the memoryless property of the exponential distribution, this is equivalent to a polling system with switch-over times between any pair of queues that are exponentially distributed with parameter

$\nu_0 = \sum_{i=1}^{N} \nu_i$ and a Markovian routing policy with routing probabilities $p_{i,j} = p_j = \nu_j/\nu_0$, $j = 1, \ldots, N$. These routing probabilities are independent of i, the index of the queue that the server has just visited. As mentioned in Section 1.3.2, the special case of a Markovian routing policy with $p_{i,j} = p_j$ (and thus also $q_j = p_j$) is also referred to as a random routing policy. Yet another equivalent interpretation is that each queue has the same back-off rate ν_0, but only activates at the end of a back-off period with an activation probability p_j. We will use this interpretation in Section 9.4.

A crucial question that we concern ourselves with in this chapter is how the back-off rates should be selected in order to minimise the overall average packet delay. To this end, we use the notation as introduced in Chapter 8 and study the equivalent optimisation problem in the polling setting. That is, for random polling systems, we study the question of which routing probabilities p_j minimise the weighted sum $\sum_{i\in\mathscr{S}} c_i \mathbb{E}[W_i]$ of mean waiting times (or equivalently, through Little's law, a weighted sum of mean queue lengths) for any set of non-negative weights $c_1, \ldots, c_N$, where $\mathscr{S} = \{1, \ldots, N\}$ as before. Of course, one can optimise all these numbers by implementing (8.21) including the set of equations (8.10) to compute $\sum_{i\in\mathscr{S}} c_i \mathbb{E}[W_i] = \sum_{i\in\mathscr{S}} \frac{c_i}{\lambda_i}(\mathbb{E}[L_i] - \rho_i)$ and searching through the complete parameter set using numerical optimisation methods. However, this method lacks transparency and provides little insight into the effects of the model parameters. Moreover, its computation time becomes prohibitively long as the number of queues increases. Therefore, there is a need for symbolic and transparent (near-)optimal expressions which are easy to implement and are suitable for optimisation purposes.

We obtain accurate approximations for the optimal routing probabilities that are expressed in closed form and are even exact when the weights are chosen such that the weighted sum represents the mean amount of (waiting) work in the system. To allow for such expressions, we assume that the first two moments of the switch-over time distributions between each pair of queues are the same, i.e. $\mathbb{E}[S_{i,j}] = \mathbb{E}[S]$ and $\mathbb{E}[S_{i,j}^2] = \mathbb{E}[S^2]$ for all $i, j \in \mathscr{S}$. Note that this is a completely valid assumption in the practice of wireless random-access networks as sketched above. Given that each queue adheres to the binomial gated or the binomial exhaustive service discipline as introduced in Section 8.2 (i.e. the branching-type service disciplines which model the network setting best), we also study the question of how to choose the exhaustiveness probabilities r_i with the same objective in mind. Contrary to the numerical method described above, the expressions that we derive provide insight into the effects of the model parameters on the waiting times and their computation times are negligible.

These (near-)optimal expressions can, however, not be used directly to obtain optimal back-off rates in the wireless random-access network setting, since these expressions involve the arrival rates of all other queues among other parameters that in practice are not known to a transmitter. Therefore, we propose a distributed algorithm that makes each node choose its back-off rate dynamically based on the durations of previous packet inter-transmissions without requiring information concerning other nodes in the network. When all nodes adhere to this algorithm, the back-off rates converge in some sense to their optimal values over time.

The remainder of this chapter is organised as follows. First, in Section 9.2, we derive expressions for the routing probabilities p_j and the exhaustiveness probabilities r_j that minimise the mean total amount of work in the system. Then, in Section 9.3, we derive approximate expressions for the same parameters that (nearly) optimise any weighted sum of the mean waiting times and conclude that these approximations are accurate by means

of a numerical study. Based on the resulting expressions for the (near-)optimal routing probabilities, we describe the algorithm for obtaining (near-)optimal back-off rates in the wireless-network setting in Section 9.4.

9.2 Minimising the mean total amount of work in the system

We start with finding the routing probabilities and exhaustiveness probabilities that minimise the mean total amount of work in the system, which is the sum of the mean amount $\sum_{i\in\mathscr{S}} \rho_i \mathbb{E}[W_i]$ of waiting work in the system and the mean amount $\sum_{i\in\mathscr{S}} \rho_i \frac{\mathbb{E}[B_i^2]}{2\mathbb{E}[B_i]}$ of remaining work to be processed of any customer that is currently being served. Since the latter expression is insensitive to the routing probabilities and the exhaustiveness probabilities, the probabilities that minimise the mean total amount of work in the system also minimise $\sum_{i\in\mathscr{S}} \rho_i \mathbb{E}[W_i]$. We therefore focus on this expression in the remainder of this section.

Recall that the server now initiates a setup to Q_j with probability p_j after a visit period regardless of which queue it actually visited. While doing so, it incurs a switch-over time with first two moments $\mathbb{E}[S]$ and $\mathbb{E}[S^2]$ for all $j \in \mathscr{S}$. Following the analysis of [54, Remark 5.4], which is based on results of [144], one can show that under these assumptions, the pseudo-conservation law in (8.24) reduces to

$$\begin{aligned}\sum_{i\in\mathscr{S}} \rho_i \mathbb{E}[W_i] = \rho \frac{\sum_{i\in\mathscr{S}} \lambda_i \mathbb{E}[B_i^2]}{2(1-\rho)} + \frac{\mathbb{E}[S]}{1-\rho} \sum_{i\in\mathscr{S}} \frac{\rho_i(1-\rho_i)}{p_i} \\ + \rho\left(\frac{\mathbb{E}[S^2]}{2\mathbb{E}[S]} - \mathbb{E}[S]\right) + \frac{\mathbb{E}[S]}{1-\rho} \sum_{i\in\mathscr{S}} \frac{\rho_i(1-\rho_i)h_i(i)}{p_i(1-h_i(i))}. \end{aligned} \tag{9.1}$$

The last term is the only term in this expression that depends on the service disciplines of the queues. Furthermore, the summands $\frac{\mathbb{E}[S]}{1-\rho} \frac{\rho_i(1-\rho_i)h_i(i)}{p_i(1-h_i(i))} = f_i(i)h_i(i)\mathbb{E}[B_i]$ of the last term equal the expected amount of work the server leaves behind at Q_i when completing a visit period there.

9.2.1 Routing probabilities

Considering (9.1), it is obvious that for any branching-type service discipline at any queue, the problem of finding the routing probabilities p_i^{opt} that minimise the mean total amount of work in the system is equivalent to the problem of finding the variable $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_N)$ that

$$\begin{aligned} \text{minimises} \quad & f(\boldsymbol{\tau}) = \sum_{i\in\mathscr{S}} \frac{\rho_i(1-\rho_i)}{\tau_i}\left(1 + \frac{h_i(i)}{1-h_i(i)}\right) = \sum_{i\in\mathscr{S}} \frac{\rho_i(1-\rho_i)}{\tau_i(1-h_i(i))} \\ \text{subject to} \quad & u(\boldsymbol{\tau}) = \sum_{i\in\mathscr{S}} \tau_i - 1 = 0,\ v_{1,j}(\boldsymbol{\tau}) = -\tau_j \le 0 \\ & \text{and } v_{2,j}(\boldsymbol{\tau}) = \tau_j - 1 \le 0 \text{ for all } j \in \mathscr{S}. \end{aligned} \tag{9.2}$$

This non-linear optimisation problem with equality and inequality constraints can be solved using a standard application of the Karush-Kuhn-Tucker conditions (see e.g. [55,

Section 5.5.3]). Let $\boldsymbol{\tau}^* = (\tau_1^*, \ldots, \tau_N^*)$ be given by $\tau_i^* = \frac{\sqrt{\rho_i(1-\rho_i)/(1-h_i(i))}}{\sum_{j\in\mathscr{S}}\sqrt{\rho_j(1-\rho_j)/(1-h_j(j))}}$. Define $\mathcal{L}$ as the number for which $\nabla f(\boldsymbol{\tau}^*) + \mathcal{L}\nabla u(\boldsymbol{\tau}^*)$ equals $\mathbf{0}$, i.e.

$$\mathcal{L} = \left(\sum_{j\in\mathscr{S}}\sqrt{\frac{\rho_j(1-\rho_j)}{1-h_j(j)}}\right)^2.$$

Furthermore, let the N-dimensional vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ be equal to $\mathbf{0}$. It is then easily verified that $\boldsymbol{\tau}^*$, $\mathcal{L}$, $\boldsymbol{a}$ and $\boldsymbol{b}$ satisfy the Karush-Kuhn-Tucker conditions

$$\begin{aligned}
&\nabla f(\boldsymbol{\tau}^*) + \mathcal{L}\nabla u(\boldsymbol{\tau}^*) + \boldsymbol{a}\nabla \boldsymbol{v}_1(\boldsymbol{\tau}^*) + \boldsymbol{b}\nabla \boldsymbol{v}_2(\boldsymbol{\tau}^*) = \mathbf{0}, && \text{(stationarity)}\\
&u(\boldsymbol{\tau}^*) = 0, \boldsymbol{v}_1(\boldsymbol{\tau}^*) \leq \mathbf{0}, \boldsymbol{v}_2(\boldsymbol{\tau}^*) \leq \mathbf{0}, && \text{(primal feasibility)}\\
&\boldsymbol{a}\boldsymbol{v}_1(\boldsymbol{\tau}^*) = \mathbf{0}, \boldsymbol{b}\boldsymbol{v}_2(\boldsymbol{\tau}^*) = \mathbf{0}, && \text{(complementary slackness)}\\
&\boldsymbol{a} \geq \mathbf{0} \text{ and } \boldsymbol{b} \geq \mathbf{0}. && \text{(non-negativity)}
\end{aligned}$$

The existence of values of $\mathcal{L}$, $\boldsymbol{a}$ and $\boldsymbol{b}$ that satisfy the Karush-Kuhn-Tucker conditions is required for $\boldsymbol{\tau}^*$ to be the solution to the optimisation problem, but it does in general not imply that $\boldsymbol{\tau}^*$ is indeed optimal. However, since the objective function $f(\boldsymbol{\tau})$ is convex in $\tau_1, \ldots, \tau_N$, these conditions are sufficient for $\boldsymbol{\tau}^*$ to be the solution to (9.2). Consequently, the optimal routing probabilities p_i that minimise the mean total amount of work in the system are given by

$$p_i^{opt} = \frac{\sqrt{\rho_i(1-\rho_i)/(1-h_i(i))}}{\sum_{j\in\mathscr{S}}\sqrt{\rho_j(1-\rho_j)/(1-h_j(j))}}. \tag{9.3}$$

REMARK 9.2.1. The optimal routing probabilities given in (9.3) generalise results obtained in [52, Section 4]. In that paper, the authors derive optimal routing probabilities for the special cases of exhaustive and gated service, i.e. $h_i(i) = 0$ and $h_i(i) = \rho_i$, respectively.

9.2.2 Exhaustiveness probabilities

We now assume that each of the queues adheres to either a binomial exhaustive or a binomial gated service discipline as described in Sections 8.1 and 8.2. We therefore partition the set $\mathscr{S}$ of queue indices in a set $\mathscr{I}_{BE}$ of indices corresponding to queues served according to the binomial exhaustive service discipline and a set $\mathscr{I}_{BG}$ of indices referring to queues with the binomial gated discipline. Recall that the last term in (9.1) is the only term in that expression that is sensitive to the service discipline and thus also to the exhaustiveness probabilities r_i. As we now have that $h_i(i) = 1-(1-\rho_i\mathbb{1}_{\{i\in\mathscr{I}_{BG}\}})r_i$, the last term of (9.1) can be simplified to $\frac{\mathbb{E}[S]}{1-\rho}\sum_{i\in\mathscr{S}} k_i(\frac{1}{r_i} - (1-\rho_i\mathbb{1}_{\{i\in\mathscr{I}_{BG}\}}))$, where

$$k_i = \frac{\rho_i(1-\rho_i)}{p_i(1-\rho_i\mathbb{1}_{\{i\in\mathscr{I}_{BG}\}})}. \tag{9.4}$$

We aim to find the exhaustiveness probabilities that minimise $\sum_{i\in\mathscr{S}}\rho_i\mathbb{E}[W_i]$. Of course, when there are no restrictions on r_i, all exhaustiveness probabilities should be chosen equal to one in order to minimise the amount of work in the system (see also

[170, Proposition 4.1]). However, as a result of this choice, the waiting times of the various customers may vary considerably depending on their time of arrival. For instance, a customer arriving just after the server concluded a visit period at his queue is likely to wait a lot longer than a customer arriving just before that time. This introduces a source of customer unfairness, especially in case of the binomial exhaustive service discipline. Furthermore, in practice, there may be costs involved in having demanding customer types (i.e. high exhaustiveness probabilities). In the wireless random-access network application, a high exhaustiveness probability of one node may heavily delay the transmission of packets by other nodes. Therefore, we add the constraint $\sum_{i\in\mathscr{S}} d_i r_i \leq 1$ to the problem, where the parameters $d_i > 0$ can be interpreted as cost parameters.

Taking everything into account, the problem of finding the optimal exhaustiveness probabilities reduces to the problem of finding the vector $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_N)$ that

$$\begin{aligned} \text{minimises} \quad & f(\boldsymbol{\tau}) = \sum_{i\in\mathscr{S}} \frac{k_i}{\tau_i} \\ \text{subject to} \quad & \upsilon_{1,i}(\boldsymbol{\tau}) = -\tau_i \leq 0,\ \upsilon_{2,i}(\boldsymbol{\tau}) = \tau_i - 1 \leq 0 \\ & \text{and } \upsilon_3(\boldsymbol{\tau}) = \sum_{j\in\mathscr{S}} d_j \tau_j - 1 \leq 0 \text{ for all } i \in \mathscr{S}. \end{aligned} \tag{9.5}$$

Note that $f(\boldsymbol{\tau})$ is a decreasing function in $\tau_1, \ldots, \tau_N$. Thus, if $\sum_{i\in\mathscr{S}} d_i < 1$, the constraint $\upsilon_3(\boldsymbol{\tau}) \leq 0$ cannot be binding, as the constraint $\upsilon_{2_i}(\boldsymbol{\tau}) \leq 0$ will prohibit that. The solution to this problem is for this case thus given by $\tau_i^* = 1$ for all $i \in \mathscr{S}$. For the case $\sum_{i\in\mathscr{S}} d_i \geq 1$, observe that if the constraints $\upsilon_{1,i}(\boldsymbol{\tau}) \leq 0$ and $\upsilon_{2,i}(\boldsymbol{\tau}) \leq 0$ did not exist, one could show that (9.5) is minimised by the vector $\boldsymbol{\tau}(0)$ with elements

$$\tau_i(0) = \frac{\sqrt{k_i/d_i}}{\sum_{j\in\mathscr{S}} \sqrt{k_j d_j}} \tag{9.6}$$

for any $i \in \mathscr{S}$. However, this vector does not necessarily satisfy the constraint $\upsilon_{2,i}(\boldsymbol{\tau}(0)) \leq 0$. It is reasonable to conjecture that if $\tau_i(0) \geq 1$, the optimal vector $\boldsymbol{\tau}^*$ satisfies $\tau_i^* = 1$. In such a case, the optimal solution may be found by truncating any values in (9.6) at one as needed, and, given that these values equal one, re-evaluating the problem to solve for the remaining values. As any of the remaining values may become larger than one after re-evaluation, this needs to be iterated until all values are not larger than one. At most N of these iterations are needed to achieve this.

To summarise all of the above, it is reasonable to conjecture that the optimal solution $\boldsymbol{\tau}^* = \boldsymbol{\tau}(N)$ to the problem specified in (9.5) has elements that are defined through the recursion

$$\begin{aligned} \tau_i(j) = {} & \mathbb{1}_{\{\tau_i(j-1)\geq 1 \vee \sum_{l\in\mathscr{S}} d_l < 1\}} \\ & + \mathbb{1}_{\{\tau_i(j-1)<1 \wedge \sum_{l\in\mathscr{S}} d_l \geq 1\}} \frac{(1 - \sum_{l\in\mathscr{S}} d_l \mathbb{1}_{\{\tau_l(j-1)\geq 1\}})\sqrt{k_i/d_i}}{\sum_{l\in\mathscr{S}} \mathbb{1}_{\{\tau_l(j-1)<1\}} \sqrt{k_l d_l}} \end{aligned} \tag{9.7}$$

for $j = 1, \ldots, N$, where (9.6) acts as an initial condition. The number j corresponds to the j-th step of the recursion. We now show that $\boldsymbol{\tau}^* = \boldsymbol{\tau}(N)$ is indeed a solution to this

problem including all mentioned constraints. To this end, we introduce

$$\mathscr{E} = \mathbb{1}_{\{\sum_{l\in\mathscr{S}} d_l\tau_l^*=1\}} \left(\frac{\sum_{l\in\mathscr{S}} \mathbb{1}_{\{\tau_l^*<1\}} \sqrt{k_l d_l}}{1-\sum_{l\in\mathscr{S}} d_l \mathbb{1}_{\{\tau_l^*\geq 1\}}} \right)^2.$$

Furthermore, let the vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ be given by $a_i = 0$ and $b_i = \mathbb{1}_{\{\tau_i^*=1\}}(k_i - d_i\mathscr{E})$, respectively. Through some straightforward computations, it can be shown that these particular choices for $\boldsymbol{a}$, $\boldsymbol{b}$ and $\mathscr{E}$ satisfy the following Karush-Kuhn-Tucker conditions for the problem in (9.5):

$$\begin{aligned}
&\nabla f(\boldsymbol{\tau}^*) + \boldsymbol{a}\nabla \boldsymbol{v}_1(\boldsymbol{\tau}^*) + \boldsymbol{b}\nabla \boldsymbol{v}_2(\boldsymbol{\tau}^*) + \mathscr{E}\nabla v_3(\boldsymbol{\tau}^*) = \boldsymbol{0}, && \text{(stationarity)}\\
&\boldsymbol{v}_1(\boldsymbol{\tau}^*) \leq \boldsymbol{0}, \boldsymbol{v}_2(\boldsymbol{\tau}^*) \leq \boldsymbol{0}, v_3(\boldsymbol{\tau}^*) \leq 0, && \text{(primal feasibility)}\\
&\boldsymbol{a}\boldsymbol{v}_1(\boldsymbol{\tau}^*) = \boldsymbol{0}, \boldsymbol{b}\boldsymbol{v}_2(\boldsymbol{\tau}^*) = \boldsymbol{0}, \mathscr{E}v_3(\boldsymbol{\tau}^*) = 0, && \text{(complementary slackness)}\\
&\boldsymbol{a} \geq \boldsymbol{0}, \boldsymbol{b} \geq \boldsymbol{0} \text{ and } \mathscr{E} \geq 0. && \text{(non-negativity)}
\end{aligned}$$

Since the Karush-Kuhn-Tucker conditions are satisfied and $f(\boldsymbol{\tau})$ is a convex function in $\tau_1, \ldots, \tau_N$, $\boldsymbol{\tau}^*$ is indeed optimal for this problem.

Going back to the original problem of finding the routing probabilities that minimise the mean total amount of work in the system under the restriction $\sum_{i\in\mathscr{S}} d_i r_i \leq 1$, we thus have that the optimal exhaustiveness probabilities r_i^{opt} are given by

$$r_i^{opt} = \tau_i(N), \tag{9.8}$$

where $\tau_i(N)$ is defined through the recursion (9.7) together with the initial value (9.6) and k_i is defined as in (9.4).

REMARK 9.2.2. In Sections 9.2.1 and 9.2.2, we have derived separate expressions for the optimal routing probabilities and exhaustiveness probabilities. Note that the found expressions for p_i^{opt} (r_i^{opt}) involve the parameters r_i (p_i), so that there is an interaction between the optimal routing probabilities and the optimal exhaustiveness probabilities. Joint optimisation of both the routing probabilities and exhaustiveness probabilities seems to be a hard problem. One may, however, obtain optimal values for both the routing probabilities and the exhaustiveness probabilities by using an alternating approach that first finds the optimal routing probabilities given an arbitrary set of exhaustiveness probabilities, then determines new optimal exhaustiveness probabilities based on the newly found routing probabilities and so on. Numerical experiments show that only a few of these iterations are already enough to obtain virtually optimal values for these parameters.

9.3 Minimising a weighted sum of mean waiting times

Now that we have found the routing probabilities and the exhaustiveness probabilities that minimise the expected amount of work in the system, the question arises which routing and exhaustiveness probabilities minimise the weighted sum $\sum_{i\in\mathscr{S}} c_i\mathbb{E}[W_i] = \sum_{i\in\mathscr{S}} \frac{c_i}{\lambda_i}(\mathbb{E}[L_i] - \rho_i)$ with arbitrary, positive weights c_i that are not necessarily equal to ρ_i. By (8.21), this sum depends on $f_i(j)$ and $f_i(i,j)$ corresponding to each $i,j \in \mathscr{S}$ and thus constitutes an intricate function of the model parameters. Optimisation of this

function is hard and does not lead to simple expressions for optimal model parameters. Therefore, we instead aim to find simple expressions that lead to a near-optimal value of $\sum_{i\in\mathscr{S}} \frac{c_i}{\lambda_i}\mathbb{E}[L_i]$, which then evidently also leads to a near-optimal value of $\sum_{i\in\mathscr{S}} c_i\mathbb{E}[W_i]$. To this end, we initially consider a more tractable problem, namely the optimisation of the weighted sum $\sum_{i\in\mathscr{S}} \frac{c_i}{\lambda_i} f_i(i)$. We thus replace $\mathbb{E}[L_i]$, the mean queue length of Q_i at any point in time, by $f_i(i)$, which refers to the mean queue length of Q_i when it is polled by the server. In Section 9.3.1, we derive expressions for routing probabilities and exhaustiveness probabilities that minimise $\sum_{i\in\mathscr{S}} \frac{c_i}{\lambda_i} f_i(i)$. Using numerical results, we will see in Section 9.3.2 that these expressions also represent probabilities that nearly optimise $\sum_{i\in\mathscr{S}} c_i\mathbb{E}[W_i]$.

9.3.1 Near-optimal expressions

We initially study the adapted problem of minimising $\sum_{i\in\mathscr{S}} \frac{c_i}{\lambda_i} f_i(i)$. Due to (8.13), we thus wish to minimise

$$\sum_{i\in\mathscr{S}} \frac{c_i}{\lambda_i} f_i(i) = \sum_{i\in\mathscr{S}} \frac{c_i\mathbb{E}[S](1-\rho_i)}{p_i(1-\rho)(1-h_i(i))}. \tag{9.9}$$

To find expressions for the routing probabilities $p_i^{n\text{-}opt}$ that minimise this sum, observe that this adapted problem is equivalent to problem (9.2), but with $\rho_i(1-\rho_i)$ in the numerator of $f(\tau)$ replaced by $c_i(1-\rho_i)$. By following the analysis of Section 9.2.1, one finds that the optimal routing probabilities for this adapted problem are given by

$$p_i^{n\text{-}opt} = \frac{\sqrt{c_i(1-\rho_i)/(1-h_i(i))}}{\sum_{j\in\mathscr{S}} \sqrt{c_j(1-\rho_j)/(1-h_j(j))}} \tag{9.10}$$

for all $i \in \mathscr{S}$.

We now consider the exhaustiveness probabilities, and we again take the constraint $\sum_{i\in\mathscr{S}} d_i r_i \leq 1$ into account. Recall that $h_i(i) = 1-(1-\rho_i\mathbb{1}_{\{i\in\mathscr{I}_{BG}\}})r_i$ when the server serves each of the queues according to the binomial exhaustive or the binomial gated service discipline. Hence, we observe by (9.9) that minimising $\sum_{i\in\mathscr{S}} \frac{c_i}{\lambda_i} f_i(i)$ is equivalent to the minimisation of $\sum_{i\in\mathscr{S}} \frac{\kappa_i}{r_i}$, where $\kappa_i = \frac{c_i(1-\rho_i)}{p_i(1-\rho_i\mathbb{1}_{\{i\in\mathscr{I}_{BG}\}})}$. By performing similar calculations to those in Section 9.2.2, we now have that the exhaustiveness probabilities $r_i^{n\text{-}opt}$ that minimise (9.9) are given by

$$r_i^{n\text{-}opt} = r_i(N), \tag{9.11}$$

where $r_i(j)$ is for all $i, j \in \mathscr{S}$ recursively defined through

$$r_i(j) = \mathbb{1}_{\{r_i(j-1)\geq 1 \vee \sum_{l\in\mathscr{S}} d_l < 1\}} + \mathbb{1}_{\{r_i(j-1)<1 \wedge \sum_{l\in\mathscr{S}} d_l \geq 1\}} \frac{(1-\sum_{l\in\mathscr{S}} d_l\mathbb{1}_{\{r_l(j-1)\geq 1\}})\sqrt{\kappa_i/d_i}}{\sum_{l\in\mathscr{S}} \mathbb{1}_{\{r_l(j-1)<1\}}\sqrt{\kappa_l d_l}}$$

with

$$r_i(0) = \frac{\sqrt{\kappa_i/d_i}}{\sum_{j\in\mathscr{S}} \sqrt{\kappa_j d_j}}.$$

We have now found the routing probabilities $p_i^{n\text{-}opt}$ and the exhaustiveness probabilities $r_i^{n\text{-}opt}$ that minimise the weighted sum $\sum_{i\in\mathscr{S}} \frac{c_i}{\lambda_i} f_i(i)$. Observe, however, that in case

TABLE 9.1: Parameter settings of the polling systems used for the numerical study of Section 9.3.2.

Parameter	Considered parameter settings
N	$2, 3, 4, 5$
Service policy	Binomial exhaustive, binomial gated
ρ	$0.1, 0.5, 0.99$
$(B_i)_{i\in\mathscr{S}}$	$\left(\text{Exponential}\left(\frac{1}{2i}\right)\right)_{i\in\mathscr{S}}, (\text{Deterministic}\,(i))_{i\in\mathscr{S}}$
$(S_i)_{i\in\mathscr{S}}$	$(\text{Uniform}\,(0,1))_{i\in\mathscr{S}}, (\text{Uniform}\,(0,100))_{i\in\mathscr{S}}$
$(\lambda_i)_{i\in\mathscr{S}}$	$\left(\frac{\rho}{N\mathbb{E}[B_i]}\right)_{i\in\mathscr{S}}, \left(\frac{2i\rho}{N(N+1)\mathbb{E}[B_i]}\right)_{i\in\mathscr{S}}, \left(\frac{2(N+1-i)\rho}{N(N+1)\mathbb{E}[B_i]}\right)_{i\in\mathscr{S}}$
$(c_i)_{i\in\mathscr{S}}$	$\left(\rho_i^2\right)_{i\in\mathscr{S}}, \left(e^{N+1-i}\right)_{i\in\mathscr{S}}$
$(d_i)_{i\in\mathscr{S}}$	$\left((N+1-i)^{-1}\right)_{i\in\mathscr{S}}, \left(\sqrt{N+1-i}\right)_{i\in\mathscr{S}}$

$c_i = \rho_i$ for all $i \in \mathscr{S}$, the expressions in (9.10) and (9.11) coincide with (9.3) and (9.8). Therefore, these expressions also represent the probabilities that minimise $\sum_{i\in\mathscr{S}} c_i\mathbb{E}[W_i]$ when $c_i = \rho_i$. Therefore, one may expect that for general c_i, the probabilities $p_i^{n\text{-}opt}$ and $r_i^{n\text{-}opt}$ nearly optimise the weighted sum of mean waiting times. In the next section, we conclude on the basis of numerical results that this is indeed the case, so that (9.10) and (9.11) can be used for the optimisation of waiting times.

9.3.2 Numerical validation

In this section, we numerically study the accuracy of the near-optimal values $p_i^{n\text{-}opt}$ and $r_i^{n\text{-}opt}$ as computed in (9.10) and (9.11). To this end, we consider a collection of 1152 model instances corresponding to all possible combinations of the parameter settings given in Table 9.1. For each of these systems, we compute the smallest possible value of the weighted sum of mean waiting times, which we denote by β^{opt}, by determining the optimal routing and exhaustiveness probabilities using numerical optimisation methods in combination with the results found in Section 8.4. We also compute the routing and exhaustiveness probabilities derived in Section 9.3.1 that should nearly optimise the weighted sum $\sum_{i\in\mathscr{S}} c_i\mathbb{E}[W_i]$ by iteratively calculating (9.10) and (9.11) using an alternating approach as sketched in Remark 9.2.2. We denote the value of the weighted sum that corresponds to these probabilities by $\beta^{n\text{-}opt}$.

Based on these numbers, we calculate the accuracy error $\Delta^{n\text{-}opt}$ of the near-optimal probabilities for each system:

$$\Delta^{n\text{-}opt} = 100\% \times \frac{\beta^{n\text{-}opt} - \beta^{opt}}{\beta^{opt}}.$$

For the sake of comparison, we also consider the baseline scenario where the routing and exhaustiveness probabilities are chosen in a naive manner, namely $p_i = \frac{1}{N}$ and $r_i = \frac{1}{d_i N}$ for all $i \in \mathscr{S}$. This leads to the weighted sum denoted by β^{base}, so that the accuracy error Δ^{base} is defined similarly to $\Delta^{n\text{-}opt}$. Note that this baseline scenario is optimal for

TABLE 9.2: The accuracy differences $\Delta^{n\text{-}opt}$ and Δ^{base} categorised in bins.

	0-0.01%	0.01-1%	1-10%	>10%
% of accuracy errors $\Delta^{n\text{-}opt}$	59.03%	31.68%	8.85%	0.43%
% of accuracy errors Δ^{base}	0.26%	7.99%	33.07%	58.68%

completely symmetric systems. In Table 9.2, the errors $\Delta^{n\text{-}opt}$ and Δ^{base} pertaining to all model instances are summarised. In particular, we see that $\Delta^{n\text{-}opt}$ is smaller than 1% in more than 90% of all cases and is even smaller than 0.01% in more than half of the cases. This suggests that the values $p_i^{n\text{-}opt}$ and $r_i^{n\text{-}opt}$ indeed virtually always lead to a weighted sum of mean waiting times that is close to optimal. They also seem to perform much better than the baseline scenario, since Table 9.2 shows almost completely opposite numbers for Δ^{base}. In particular, the accuracy errors of the baseline scenario are larger than 1% in more than 90% of all cases and they even exceed 10% in more than half of the cases. This effect is also captured by the fact that the average value of $\Delta^{n\text{-}opt}$ equals 0.425% and that of Δ^{base} equals 24.18%.

To give insight into parameter effects, Table 9.3 displays average values of $\Delta^{n\text{-}opt}$ categorised in some of the model parameters. From Table 9.3(a), we conclude that the accuracy of the near-optimal values is hardly influenced by the number of queues in the system. However, judging by Table 9.3(b), the accuracy is sensitive to the load ρ offered to the server. As any choice for the routing and exhaustiveness probabilities is optimal in case of a zero load, it makes sense that the accuracy degrades slowly when the load increases. Table 9.3(c) suggests that the near-optimal values tend to perform better when there is less stochasticity in the system. Judging by Table 9.3(d), the performance is also increasing in the average duration of the switch-over times. This can be explained by the fact that routing probabilities or exhaustiveness probabilities have less impact on the waiting time when the switch-over times become an increasing source of waiting time. Finally, Tables 9.3(e) and 9.3(f) suggest that a higher level of asymmetry in the model parameters leads to larger inaccuracies. This is as expected, since the near-optimal probabilities are optimal when the system to be optimised is completely symmetric.

REMARK 9.3.1. In Sections 9.2.2 and 9.3.1, we have derived expressions for exhaustiveness probabilities that (nearly) optimise a weighted sum of the mean waiting times. However, in practice, one may also be interested in keeping the level of variation in the waiting times low. In an effort to reduce the level of variation, one may thus choose to adapt the exhaustiveness probability in a dynamic fashion at the start of every n-th visit period at that queue, based on the number of customers present in the queue at that particular polling epoch. More specifically, let r_i^{opt} be the expression of the (near-)optimal exhaustiveness probability at Q_i as found before, and let $f_{i,r_i^{opt}}(i)$ be the corresponding expected queue length at Q_i at the start of any visit period to Q_i, which can be computed through (8.13). By using (8.16), we find that the expected number of customers that the server leaves behind at that queue when initiating the next switch-over period is given by $f_{i,r_i^{opt}}(i)h_i(i) = f_{i,r_i^{opt}}(i)\left(1-(1-\rho_i\mathbb{1}_{\{i\in\mathscr{S}_{BG}\}})r_i^{opt}\right)$. Likewise, if one decides that the server at Q_i should adhere to the exhaustiveness probability $r_{i,n}^{dyn}$ instead during the n-th visit period at Q_i, at the start of which $z_{i,n}$ customers are present, the expected

TABLE 9.3: Average accuracy error categorised in some of the model parameters as specified in Table 9.1.

(a)

N	2	3	4	5
Average $\Delta^{n\text{-}opt}$	0.41%	0.43%	0.43%	0.43%

(b)

ρ	0.1	0.5	0.99
Average $\Delta^{n\text{-}opt}$	0.00%	0.03%	1.24%

(c)

$(B_i)_{i\in\mathscr{S}}$	$\left(\text{Exponential}\left(\frac{1}{2i}\right)\right)_{i\in\mathscr{S}}$	$(\text{Deterministic}\,(i))_{i\in\mathscr{S}}$
Average $\Delta^{n\text{-}opt}$	0.66%	0.19%

(d)

$(S_i)_{i\in\mathscr{S}}$	$(\text{Uniform}\,(0,1))_{i\in\mathscr{S}}$	$(\text{Uniform}\,(0,100))_{i\in\mathscr{S}}$
Average $\Delta^{n\text{-}opt}$	0.61%	0.24%

(e)

$(c_i)_{i\in\mathscr{S}}$	$\left(\rho_i^2\right)_{i\in\mathscr{S}}$	$\left(e^{N+1-i}\right)_{i\in\mathscr{S}}$
Average $\Delta^{n\text{-}opt}$	0.23%	0.62%

(f)

$(d_i)_{i\in\mathscr{S}}$	$\left((N+1-i)^{-1}\right)_{i\in\mathscr{S}}$	$\left(\sqrt{N+1-i}\right)_{i\in\mathscr{S}}$
Average $\Delta^{n\text{-}opt}$	0.62%	0.23%

number of customers left behind at the start of the subsequent switch-over period equals $z_{i,n}\left(1-(1-\rho_i\mathbb{1}_{\{i\in\mathscr{I}_{BG}\}})r_{i,n}^{dyn}\right)$. To reduce variation in the waiting times, $r_{i,n}^{dyn}$ could thus be chosen such that these two numbers are the same:

$$z_{i,n}(1-(1-\rho_i\mathbb{1}_{\{i\in\mathscr{I}_{BG}\}})r_{i,n}^{dyn}) = f_{i,r_i^{opt}}(i)\left(1-\left(1-\rho_i\mathbb{1}_{\{i\in\mathscr{I}_{BG}\}}\right)r_i^{opt}\right).$$

By rewriting this equation and observing that $r_{i,n}^{dyn}$ cannot drop below zero or exceed one, we have that

$$r_{i,n}^{dyn} = \left(\min\left\{1,(1-\rho_i\mathbb{1}_{\{i\in\mathscr{I}_{BG}\}})^{-1}\left(1-\frac{f_{i,r_i^{opt}}(i)}{z_{i,n}}\left(1-(1-\rho_i\mathbb{1}_{\{i\in\mathscr{I}_{BG}\}})\right)r_i^{opt}\right)\right\}\right)^+.$$

Observe that this expression only depends on model parameters that pertain to Q_i and not to other queues. Choosing the exhaustiveness probabilities dynamically in this way

makes customers waiting in a longer (shorter) queue than average have a higher (lower) probability of getting served during the current visit period than in the static case. This evidently reduces the variance of the waiting times. There is, however, no guarantee that the mean waiting times $\mathbb{E}[W_i]$ will not increase as a result.

9.4 A distributed algorithm for wireless random-access networks

Up to now, we have derived expressions for certain model parameters that are optimal or nearly optimal in some sense. Among them, there are expressions for the routing probabilities p_j that (nearly) optimise a weighted sum of the mean waiting times (or equivalently, mean queue lengths) under the assumption of random routing. As seen in (9.3) and (9.10), these expressions are of the form

$$p_i^{opt} = \frac{\gamma_i}{\sum_{j\in\mathscr{S}} \gamma_j}, \tag{9.12}$$

where the coefficients γ_i are positive and only depend on parameters pertaining to Q_i. Hence, the numerator of (9.12) only depends on Q_i-specific values, but the denominator pertains to parameters of all queues for normalisation purposes.

We now consider the wireless random-access network setting as described in Section 9.1, and as mentioned there, we assume that each node has the same back-off rate ν_0, but only activates at the end of a back-off period with an activation probability p_j. An important question is what the activation probability of each node should be in order to minimise the overall mean number of packets waiting to be transmitted and hence the overall mean delay. Although each of the nodes in the network operates autonomously, it is reasonable to assume that the nodes are cooperative and in principle strive to achieve such a common goal. The found expressions of the form given in (9.12) in principle offer a solution to this type of problem. However, these expressions are not directly applicable to the wireless setting. Recall that the nodes operate in a distributed way. In other words, they operate concurrently on the basis of the partial information that is known to them. The information known to each node includes the value γ_i and the observed inter-transmission times so far, but not the values γ_j pertaining to other nodes.

To overcome this problem, we propose an algorithm that makes each node update its activation probability in such a way that these probabilities tend towards their (nearly) optimal values $\frac{\gamma_i}{\sum_{j\in\mathscr{S}} \gamma_j}$, provided that all nodes in the network follow this algorithm. The algorithm works in a distributed fashion as desired: all the nodes execute this algorithm concurrently, but autonomously based on inter-transmission times observed thus far and their value of γ_i. In Section 9.4.1, we describe two possible variants of the algorithm. The first variant makes the nodes choose activation probabilities that over time converge with probability one to (values near) the desired values p_i^{opt}. Although in the second variant the activation probabilities converge to their limiting values in a weaker sense, we will see that this variant is more robust to a variable population of nodes in the network or changing values of γ_i. Section 9.4.2 subsequently examines both variants of the algorithm in more detail and elaborates on their convergence properties. Finally, we provide numerical examples for both variants in Section 9.4.3.

9.4.1 Description of the distributed algorithm

We now propose an algorithm, which prescribes for each node which activation probability it should adopt based on the information available to that specific node. We assume that the information known to any of the N present nodes, which we index by i, includes the value γ_i and the durations of the previous inter-transmission times. First, we introduce some additional notation. We index time by n, so that $X(n)$ refers to the duration of the n-th inter-transmission time. The activation probability of node i during the n-th inter-transmission time is denoted by $p_i(n)$. As the back-off rates of the nodes each equal ν_0, the inter-transmission time $X(n)$ is exponentially distributed with rate $\nu_0 \sum_{j\in\mathscr{S}} p_j(n)$, where the set $\mathscr{S} = \{1,\ldots,N\}$ represents the N nodes present in the network. Finally, for the sake of conciseness, we use $[x]_y^z$ as shorthand notation for $\min\{\max\{x,y\},z\}$.

Now that the required additional notation has been introduced, we proceed to describe a distributed algorithm that makes the activation probabilities move towards their (near-)optimal values in the long run.

ALGORITHM 9.4.1. *Let α and M be positive constant coefficients, $\alpha < \gamma_i^{-1}$. Furthermore, let the i-th node ($i \in \mathscr{S}$) have an initial activation probability of $p_i(1) = \gamma_i\theta_i(0)$, $i = 1,\ldots,N$, where $\theta_i(0)$ is assumed to be in the interval $[\alpha,\gamma_i^{-1}]$. After the n-th transmission time, it calculates*

$$\theta_i(n) = [(1-\epsilon(n))\theta_i(n-1) + \epsilon(n)M(\nu_0 X(n) - 1)]_\alpha^{\gamma_i^{-1}}, \tag{9.13}$$

where the $\epsilon(n)$ are step sizes that depend on n. Subsequently, node i updates its activation probability according to

$$p_i(n+1) = \gamma_i\theta_i(n). \tag{9.14}$$

When each node adheres to this algorithm, the activation probabilities $p_i(n)$ of the various nodes will eventually converge in some sense (depending on the choice of $\epsilon(n)$) to the value $\gamma_i\hat{\theta}$, where $\hat{\theta}$ is given by

$$\hat{\theta} = \frac{-M}{2} + \sqrt{\frac{M^2}{4} + \frac{M}{\sum_{j\in\mathscr{S}}\gamma_j}}, \tag{9.15}$$

provided that $\alpha < \hat{\theta}$.

In Section 9.4.2, we examine this algorithm in detail and focus on the convergence properties of this algorithm. However, we first study this algorithm to see why it forms a solution to our problem and to explore the roles of the step sizes and the algorithm's coefficients. To this end, we observe that if the values $X(n)$ did not exhibit random noise, i.e. $X(n) = \mathbb{E}[X(n)] = \left(\sum_{j\in\mathscr{S}}\gamma_j\nu_0\theta_j(n-1)\right)^{-1}$, the N-dimensional difference equation in (9.13) would reduce to

$$\theta_i(n) = \left[(1-\epsilon(n))\theta_i(n-1) + \epsilon(n)M\left(\frac{1}{\sum_{j\in\mathscr{S}}\gamma_j\theta_j(n-1)} - 1\right)\right]_\alpha^{\gamma_i^{-1}}, \tag{9.16}$$

for each $i \in \mathscr{S}$. In this N-dimensional difference equation, each of the $\theta_i(n)$ evolves in exactly the same way, so that the fixed point $\boldsymbol{\theta}$ of this N-dimensional difference equation must satisfy $\theta_i = \hat{\theta}$ for all i for some value $\hat{\theta}$ as a result of symmetry. Thus, the problem of

finding the (positive) fixed point of the N-dimensional difference equation can be reduced to finding the (positive) solution of the one-dimensional problem

$$\hat{\theta} = \left[(1-\epsilon(n))\hat{\theta} + \epsilon(n) M \left(\frac{1}{\hat{\theta} \sum_{j \in \mathscr{S}} \gamma_j} - 1 \right) \right]_{\alpha}^{\gamma_i^{-1}},$$

which is easily seen to be given by $\hat{\theta}$ as specified in (9.15) when α is smaller than the expression displayed in (9.15). Furthermore, it is easily verified that this expression is a unique fixed point of (9.16) and tends to $\frac{1}{\sum_{j \in \mathscr{S}} \gamma_j}$ when $M \to \infty$. By this observation and (9.14), it is thus not surprising that the activation probabilities p_i of the nodes will eventually be close to their (near-)optimal values p_i^{opt} when taking M large enough. In fact, as M tends to infinity, the expression of (9.15) tends to its limit from below. This is a desired property, as the sum of the activation probabilities does not exceed one in that case.

We consider two different variants of this algorithm. The first variant uses step sizes $\epsilon(n)$ that satisfy the conditions

$$\begin{aligned} &\epsilon(n) \geq 0 \text{ for all } n \geq 1, \epsilon(n) \to 0 \text{ if } n \to \infty, \\ &\sum_{n=1}^{\infty} \epsilon(n) = \infty \text{ and } \sum_{n=1}^{\infty} (\epsilon(n))^2 < \infty. \end{aligned} \tag{9.17}$$

As we will see in Section 9.4.2, the activation probabilities converge with probability one to $\gamma_i \hat{\theta}$ when using this variant. We also study a second variant of the algorithm, namely the one which assumes that the step sizes $\epsilon(n) = \epsilon$ are constant over time. We will see in Section 9.4.2 that although stationary iterates of (9.13) will then still be contained in a small area around $\gamma_i \hat{\theta}$, this variant does not exhibit convergence with probability one, since the step sizes do not decrease over time. Due to the constant step sizes, however, the second variant is more suitable for use in networks with a variable population of nodes or changing values of γ_i, i.e. for settings for which the (near-)optimal activation probability p_i^{opt} is of a variable nature. In the first variant, convergence of the activation probabilities to new values of p_i^{opt} would after some point become unacceptably slow due to the decreasing step sizes. The second variant does not have this problem.

When deploying the algorithm, it is important to choose the coefficients of the algorithm well. In particular, the lower bound α needs to be chosen positive so as to keep the algorithm from producing negative control parameters $\theta_i(n)$, but smaller than $\hat{\theta}$ so as to preserve the desired limiting values. Due to the bounds α and γ_i^{-1}, the control parameters $\boldsymbol{\theta}(n)$ take values in the hypercube $\mathscr{H} = \{\boldsymbol{\theta} : \boldsymbol{\theta} \in \mathbb{R}^{|\mathscr{S}|} \wedge \alpha \leq \theta_i \leq \gamma_i^{-1} \ \forall i \in \mathscr{S}\}$. As for the coefficient M, we have already seen that the higher the value of M, the closer the limiting value $\gamma_i \hat{\theta}$ is to the desired value p_i^{opt}. However, a large M also implies that the iterates of (9.13) are prone to a significant amount of random noise. To prevent this, the step sizes should be chosen such that $\epsilon(n)M$ (or in case of the second variant, ϵM) is small enough. Observe that the step sizes should not be taken too small either, as this will result in slow convergence.

9.4.2 Convergence properties

Now that Algorithm 9.4.1 has been introduced properly, we study this algorithm in detail and establish the convergence properties of the two variants as considered in the previous section. Although both variants exhibit a different form of convergence, we will see that the arguments needed to establish these convergence properties are similar. In particular, results from [153] imply that the limiting result in both variants coincides with the unique asymptotically stable point of the same N-dimensional ordinary differential equation, which can informally be thought of as the continuous-time equivalent of (9.13). To be more specific, we can rewrite (9.13) in the form

$$\theta_i(n) = \theta_i(n-1) + \epsilon(n)Y_i(n) + \epsilon(n)Z_i(n) \tag{9.18}$$

for each $i \in \mathscr{S}$, where the variables $Y_i(n)$ and $Z_i(n)$ are given by

$$Y_i(n) = M(\nu_0 X(n) - 1) - \theta_i(n-1)$$

and

$$Z_i(n) = \left(\frac{\alpha - \theta_i(n-1)}{\epsilon(n)} - Y_i(n)\right)^+ + \left(\frac{\gamma_i^{-1} - \theta_i(n-1)}{\epsilon(n)} - Y_i(n)\right)^-,$$

respectively. Thus, $Z_i(n)$ is the number with the smallest absolute value needed to keep $\theta_i(n+1)$ between α and γ_i^{-1}. The N-dimensional ordinary differential equation referred to in [153] can now be expressed as

$$\dot{\theta}_i = g_i(\boldsymbol{\theta}) + z_i(\boldsymbol{\theta}) \tag{9.19}$$

for all $i \in \mathscr{S}$, where $\boldsymbol{\theta}$ is a function of a continuous-time parameter t rather than the discrete-time parameter n as before. We use $\dot{\theta}_i$ and $\dot{f}(\boldsymbol{\theta})$ to represent the derivative of θ_i or any function $f(\boldsymbol{\theta})$, respectively, with respect to this continuous-time parameter. The function $g_i(\boldsymbol{\theta})$ is given by

$$g_i(\boldsymbol{\theta}) = \mathbb{E}[Y_i(n) \mid \boldsymbol{\theta}(n-1) = \boldsymbol{\theta}] = M\left(\left(\sum_{j\in\mathscr{S}} \gamma_j \theta_j\right)^{-1} - 1\right) - \theta_i.$$

Furthermore, $z_i(\boldsymbol{\theta})$ is again a number with the smallest absolute value needed to keep $\boldsymbol{\theta}$ from leaving the hypercube $\mathscr{H}$. Thus, $z_i(\boldsymbol{\theta})$ becomes positive (negative) whenever θ_i takes the boundary value of α (γ_i^{-1}) and needs to be 'pushed' back for $\boldsymbol{\theta}$ to stay in $\mathscr{H}$. More specifically, we have that

$$z_i(\boldsymbol{\theta}) = -g_i(\boldsymbol{\theta}) \mathbb{1}_{\{(\theta_i = \alpha \wedge g_i(\boldsymbol{\theta}) < 0) \vee (\theta_i = \gamma_i^{-1} \wedge g_i(\boldsymbol{\theta}) > 0)\}}.$$

To find the asymptotically stable points of (9.19), we first look for fixed points of (9.19), i.e. points for which $\dot{\theta}_i = 0$ for all $i \in \mathscr{S}$. To this end, note that $g_i(\boldsymbol{\theta})$ has a positive root $\boldsymbol{\theta}^*$ with elements given by $\theta_i^* = \hat{\theta}$ for all $i \in \mathscr{S}$ (cf. (9.15)), provided that $\alpha < \hat{\theta}$. As a result, $g_i(\boldsymbol{\theta})$ is contained in the interior of $\mathscr{H}$. Since $g_i(\boldsymbol{\theta})$ is decreasing in θ_i, $g_i(\boldsymbol{\theta})$ is positive when θ_i equals α, as this is a lower boundary of $\mathscr{H}$. Similarly, $g_i(\boldsymbol{\theta})$ is negative when θ_i equals the upper boundary γ_i^{-1}. As a result, we have that $z_i(\boldsymbol{\theta}) = 0$ for any $i \in \mathscr{S}$ and $\boldsymbol{\theta} \in \mathscr{H}$. Thus, any fixed point $\boldsymbol{\theta}^*$ of (9.19) satisfies $g_i(\boldsymbol{\theta}^*) = 0$ for all

$i \in \mathscr{S}$. Consequently, $\boldsymbol{\theta}^* = (\theta_1^*, \ldots, \theta_N^*) = (\hat{\theta}, \ldots, \hat{\theta})$ is a fixed point of (9.19). This fixed point is moreover unique, as $g_i(\boldsymbol{\theta})$ only has one positive root due to its decreasingness in the non-negative orthant.

In order to apply the results from [153], it remains to be shown that the unique fixed point $\boldsymbol{\theta}^*$ is asymptotically stable. To this end, we consider the Lyapunov function

$$L(\boldsymbol{\theta}) = \left(\max_{i \in \mathscr{S}} \{\theta_i\} - \min_{j \in \mathscr{S}} \{\theta_j\} \right)^2 + \left(\sum_{k \in \mathscr{S}} \gamma_k (\theta_k - \theta_k^*) \right)^2. \tag{9.20}$$

It is evident that $L(\boldsymbol{\theta}^*) = 0$ and $L(\boldsymbol{\theta}) > 0$ for all $\boldsymbol{\theta} \in \mathscr{H} \backslash \{\boldsymbol{\theta}^*\}$. Furthermore, we see that the time-derivative of $L(\boldsymbol{\theta})$ satisfies

$$\begin{aligned} \dot{L}(\boldsymbol{\theta}) &= 2(\dot{\theta}_{\arg\max_{i \in \mathscr{S}} \{\theta_i\}} - \dot{\theta}_{\arg\min_{j \in \mathscr{S}} \{\theta_j\}}) \left(\max_{i \in \mathscr{S}} \{\theta_i\} - \min_{j \in \mathscr{S}} \{\theta_j\} \right) + 2 \sum_{k \in \mathscr{S}} \gamma_k \dot{\theta}_k \sum_{l \in \mathscr{S}} \gamma_l (\theta_l - \theta_l^*) \\ &= -2 \left(\max_{i \in \mathscr{S}} \{\theta_i\} - \min_{j \in \mathscr{S}} \{\theta_j\} \right)^2 + 2 \sum_{k \in \mathscr{S}} \gamma_k g_k(\boldsymbol{\theta}) \sum_{l \in \mathscr{S}} \gamma_l (\theta_l - \theta_l^*), \end{aligned} \tag{9.21}$$

where the second equality follows from (9.19) and the fact that $z_i(\boldsymbol{\theta})$ equals zero for all $\boldsymbol{\theta} \in \mathscr{H}$. Note that the first term of the right-hand side of (9.21) is negative, except when $\theta_i = \theta_j$ for all $i, j \in \mathscr{S}$, in which case the first term equals zero. As for the second term, observe that any $\boldsymbol{\theta} \in \mathscr{H}$ that satisfies $\sum_{l \in \mathscr{S}} \gamma_l \theta_l = \sum_{l \in \mathscr{S}} \gamma_l \theta_l^*$ is a root of $\sum_{k \in \mathscr{S}} \gamma_k g_k(\boldsymbol{\theta})$. As $\sum_{k \in \mathscr{S}} \gamma_k g_k(\boldsymbol{\theta})$ is decreasing in $\sum_{l \in \mathscr{S}} \gamma_l \theta_l$, it thus follows that the second term is negative, except when $\sum_{l \in \mathscr{S}} \gamma_l \theta_l = \sum_{l \in \mathscr{S}} \gamma_l \theta_l^*$. Combining these observations, we have that $\dot{L}(\boldsymbol{\theta}^*) = 0$ and $\dot{L}(\boldsymbol{\theta}) < 0$ for all $\boldsymbol{\theta} \in \mathscr{H} \backslash \{\boldsymbol{\theta}^*\}$. By standard theory on Lyapunov functions (see e.g. [139]) and the properties of the particular Lyapunov function $L(\boldsymbol{\theta})$ as established above, we conclude that the fixed point $\boldsymbol{\theta}^*$ is asymptotically stable.

Now that we have identified the unique asymptotically stable point of (9.19), we can apply the results from [153] to obtain the convergence properties of both variants of the algorithm. As proved in [153, Theorem 5.2.1], the iterates of (9.13) (or (9.18)) converge under very broad assumptions (which are satisfied here) with probability one to the asymptotically stable point $\boldsymbol{\theta}^* = (\hat{\theta}, \ldots, \hat{\theta})$ of (9.19), in case the step sizes $\epsilon(n)$ decay over time subject to the conditions given in (9.17). Thus, in the first variant of the algorithm, the activation probabilities $p_i(n)$ converge with probability one to the value $\gamma_i \theta_i^*$ for all $i \in \mathscr{S}$, which we have already seen to be close to the desired value p_i^{opt}.

As for the second variant, it is stated in [153, Theorem 8.2.1] that for similar algorithms with constant step sizes, the iterates of (9.13) will not converge with probability one anymore, but will still in the long run fluctuate around the asymptotically stable point $\boldsymbol{\theta}^*$ of the ordinary differential equation (9.19). More specifically, the theorem implies there always exists an $\epsilon > 0$ small enough so that the probability that a stationary value $\theta_i(n)$ is contained in any arbitrarily small area around this fixed point exceeds any given positive value smaller than one. Thus, the $\theta_i(n)$ converge to the same limiting values as in the first variant, but in a weaker sense. However, as discussed in Section 9.4.1, the second variant can handle changing values of $\sum_{j \in \mathscr{S}} \gamma_j$ better due to the constant step size.

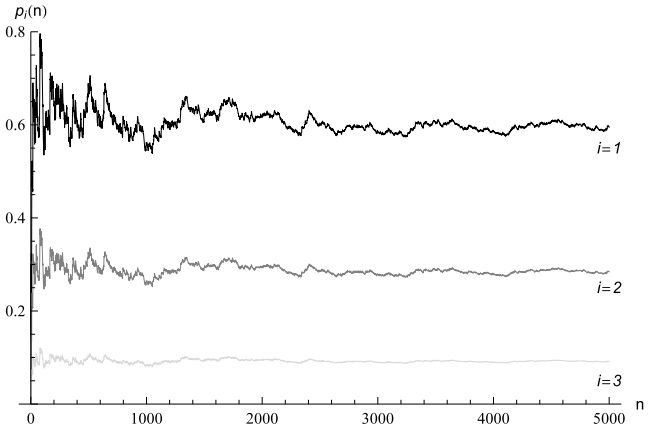

FIGURE 9.1: Evolution of the activation probabilities in the first example.

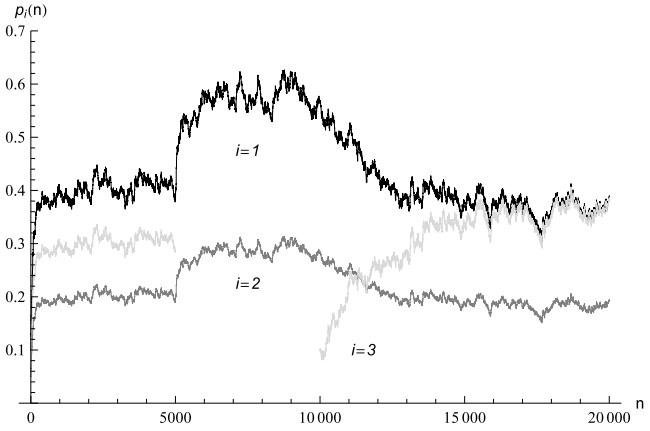

FIGURE 9.2: Evolution of the activation probabilities in the second example.

9.4.3 Numerical examples

We end the study of the distributed algorithm with two numerical examples illustrating the discussed variants of Algorithm 9.4.1. First, we consider three interfering nodes in a network with $\gamma_1 = \theta_1(0) = 0.3$, $\gamma_2 = \theta_2(0) = 0.15$ and $\gamma_3 = \theta_3(0) = 0.05$. To control their activation probabilities, the nodes adopt the first variant of the algorithm. The coefficients of the algorithm are given by $\alpha = 1/1000$ and $M = 100$. Furthermore, the step sizes are chosen according to $\epsilon(n) = (n + \log(n) + 10M)^{-1}$ and as a result satisfy the conditions in (9.17). Figure 9.1 plots the activation probabilities as generated by the three nodes adhering to the algorithm with these settings. As expected, the three back-off rates converge to (values close to) their optimal values $p_1^{opt} = 0.6$, $p_2^{opt} = 0.3$ and $p_3^{opt} = 0.1$. Furthermore, the back-off rates become less volatile as time progresses due to the decaying step sizes.

To illustrate the second variant of the distributed algorithm, we again consider three

nodes, this time with $\gamma_1 = \theta_1(0) = 0.1$, $\gamma_2 = \theta_2(0) = 0.05$ and $\gamma_3 = \theta_3(0) = 0.075$, respectively. To show that the second variant allows for changing settings in the network, we assume that after 5000 packet inter-transmissions, the third node disappears from the network. Furthermore, after 10000 packet inter-transmissions a new third node appears, this time with parameter $\gamma_3 = \theta_3(0) = 0.1$. To control the activation probabilities of the various nodes over time, we adopt Algorithm 9.4.1 with coefficients $\alpha = 1/1000$, $M = 35$ and constant step sizes $\epsilon(n) = \epsilon = 1/2000$.

Figure 9.2 plots the resulting evolution of the various activation probabilities $p_i(n)$. Initially, the activation probabilities fluctuate around values that are slightly smaller than the optimal activation probabilities $p_1^{opt} = 4/9$, $p_2^{opt} = 2/9$ and $p_3^{opt} = 1/3$. Thus, the sum of the activation probabilities rarely exceeds one, as desired. When the third node disappears after 5000 packet inter-transmissions, the activation probabilities of the remaining nodes adapt to the new situation and correctly move towards new limiting values. When a new third node appears after the 10000-th packet inter-transmission, the activation probabilities once again adjust to the new situation. In particular, we see that the activation probabilities of the first and third node eventually coincide, since $\gamma_1 = \gamma_3$ for $n \geq 10000$.

REMARK 9.4.1. In practice, it may happen that a node has no packets to transmit. In such a case, the 'empty' node will deactivate immediately after activation due to its lack of packets to be transmitted. However, other nodes might not be able to detect such an activation followed by an immediate deactivation. This would then result in the (de)activating node updating its control parameter $\theta_i(n)$, while the other nodes do not update their control parameters. This may cause problems, as the algorithm requires all of the nodes to update their control parameters simultaneously. To avoid these problems, one may adapt the algorithm such that a node now sets its own activation probability equal to zero when it has no packets to transmit. Otherwise, it sets its activation probability as before according to the original Algorithm 9.4.1. Simply put, an 'empty' node no longer activates if it has no packets to transmit. This minor adjustment has the advantage that, when certain nodes remain empty for a larger amount of time, the activation probabilities of the other nodes will adapt to this situation accordingly.

Part III

The carousel storage model

10

THE CYCLIC CAROUSEL STORAGE MODEL

In this chapter, we initiate the analysis of the final layered queueing network that we study in this dissertation, namely the carousel storage model as introduced in Section 1.3.3. We first study the waiting times of the server in case the server polls the stations in a cyclic order. Under this cyclic assumption, we give a sufficient condition for the existence of a limiting waiting-time distribution and we study the tail behaviour of this distribution. Furthermore, assuming that preparation times are exponentially distributed, we give a detailed description of the resulting discrete-time Markov chain that leads to the limiting waiting-time distribution. Finally, we provide extensive numerical results investigating the effect of the system parameters to the waiting time of the server.

10.1 Introduction

The carousel storage model, which we study in this chapter, in fact constitutes a polling model, but it differs substantially from the type of polling models studied in previous chapters. There is now an infinite number of customers waiting at each of the queues, and the server now serves at most one customer per visit period. The most important difference, however, lies in the fact that the carousel storage model has the added feature of customers undergoing a preparation phase before they are ready to be served by the server. As a result, when visiting a queue, the server may have to wait for the customer overthere to have his preparation phase finished before the actual service can be provided. As explained in Section 1.3.3, the server thus becomes a customer himself in some sense, which is why this model fits the framework of layered queueing networks.

We concern ourselves with the waiting times of the server under the assumption that the server visits the stations in a fixed, cyclic order. Under this assumption, this model leads to a Lindley-type equation, which for two service stations evaluates to

$$W \stackrel{d}{=} (B - A - W)^+ .$$

Here, B denotes the preparation time, A denotes the service time and W is the waiting time of the server. The difference from the original Lindley equation (cf. [161]) is the minus sign in front of W at the right-hand side of the equation, which in Lindley's equation is a plus. Lindley's equation describes the waiting time of a customer in a single-server

queue. It is one of the fundamental and best-studied equations in queueing theory. For a detailed study on Lindley's equation, we refer to [19, 67] and the references therein. The implications of this 'minor' difference in sign are rather far-reaching. Lindley's equation has a simple solution, whereas the equivalent equation with the minus sign (cf. (10.1)) is very challenging to solve without making additional assumptions, even for the case of two service stations (cf. [264]).

This adapted Lindley-type equation surprisingly emerges when studying maximum-weight independent sets in sparse random graphs. More specifically, consider an n-node sparse random (potentially regular) graph and let the nodes of the graph be equipped with non-negative weights, independently generated according to some common distribution. Rather than only the size of the maximum independent set, consider also the maximum *weight* of an independent set. It is shown in [103] that for certain weight distributions, a limiting result can be proved both for the maximum independent set and the maximum weight independent set. What is crucial in this computation is the Lindley-type equation (10.2) (cf. [103, Equation (3)]). This recursion provides a surprising link between queueing theory and random graphs.

At a glance, other than the analytical results, the major insights that we gain for the 'cyclic' carousel storage model in this chapter are as follows. First, we observe that any variability in preparation times has a greater influence on the system's performance than the variability in service times. Thus, in the healthcare setting mentioned in Section 1.3.3, one could say that it pays more to have a reliable nurse than a reliable specialist. Second, a *small* variability of preparation times actually improves the performance of the server under cyclic routing assumptions, in the sense that he waits less frequently (cf. Figure 10.2). However, it also decreases the throughput. Thus, the system's designer may wish to consider how to balance these conflicting goals. Next, when deciding how many stations to assign to a server in the cyclic model, the shape of the distribution plays a role. However, in general, when preparation times are smaller than service times and the variability in the preparation times is low, only few stations per server (about 5 or 6) already come close to the optimal throughput. The last major insight that we gain is of a mathematical nature. We observe that as the number of stations goes to infinity, the waiting times of the server become uncorrelated. The correlation structure of the waiting times, however, turns out to be very surprising. We additionally provide an analytic lower bound on the throughput for the cyclic case and an empirical upper bound. Both of these bounds are easy to compute, converge exponentially to the true throughput as the number of stations goes to infinity and are tight in some cases. Thus, we get quick and accurate estimates on the system's performance.

The rest of the chapter is organised as follows. The notation for the cyclic carousel storage model is presented in Section 10.2. In Section 10.3, we provide analytical results for the waiting time of the server. More specifically, we give a sufficient condition for the existence of a limiting waiting-time distribution and investigate its tail behaviour. Under the assumption that preparation times are exponential, we also study the transient behaviour of the waiting time and provide the transition matrix of the underlying discrete-time Markov chain. Finally, Section 10.4 provides a thorough treatment of the insights that we described above concerning the system's performance for the cyclic model. In the next chapter, we will extensively study the question of how these insights change when we drop the assumption of cyclic service.

10.2 Model description and notation

We assume that there are $N \geq 2$ identical service stations, $Q_1, \ldots, Q_N$, operated by a single server. Each of these service stations has an infinite supply of customers. Before being served by the server for a duration A, a customer must first undergo a preparation phase with duration B (not involving the server). Thus, the server, after having finished serving a customer at one station, may have to wait for the preparation phase of the customer at the next station to be completed. Immediately after the server concludes his service at some station, another customer from the queue begins his preparation phase there while the server moves to the next station. Consequently, at each point in time, there is exactly one customer at a service station who is either in service, waiting for service or undergoing preparation. Unless otherwise stated, we assume that A and B are continuous random variables with finite means, general distribution functions F_A (F_B) and Laplace-Stieltjes transforms $\widetilde{A}(s) = \mathbb{E}[e^{-sA}]$ and $\widetilde{B}(s) = \mathbb{E}[e^{-sB}]$.

In this chapter, we are concerned with the waiting time of the server when assuming he serves the stations in a cyclic order. Thus, after having served a customer at service station Q_i, the server will move to service station Q_{i+1} to serve a customer there. Note that indices of service stations are to be understood modulo N, so that service station Q_i actually refers to service station $Q_{((i-1) \bmod N)+1}$. We will refer to this as the *cyclic model* or the *cyclic case*. In Chapter 11, we compare this model to the so-called *dynamic model*, where the server does not necessarily poll the service stations in a cyclic order, but always visits the service station corresponding to the customer that finishes or has finished its preparation phase the earliest.

The waiting time incurred by the server in the cyclic model can be characterised as follows. Let B_n denote the preparation time of the n-th customer served, and let A_n be the time the server spends on this customer. We assume that $\{B_n\}_{n\geq 1}$ and $\{A_n\}_{n\geq 1}$ are comprised of independent and identically distributed realisations of the random variables B and A. The waiting time W_n^C incurred by the server just before serving the n-th customer then satisfies the equation

$$W_{n+1}^C = \left(B_{n+1} - \sum_{i=n-N+2}^{n} A_i - \sum_{i=n-N+2}^{n} W_i^C \right)^+. \tag{10.1}$$

This equation can be rewritten as

$$W_{n+1}^C = \left(X_{n+1} - \sum_{i=n-N+2}^{n} W_i^C \right)^+, \tag{10.2}$$

where $X_{n+1} = B_{n+1} - \sum_{i=n-N+2}^{n} A_i$. Note that $\{X_n, n \geq N-1\}$ is comprised of identically distributed realisations of a random variable X. However, these realisations are not necessarily independent. They are only independent with an $(N-1)$-lag. Thus, for example, X_N, X_{2N-1}, X_{3N-2}, $X_{4N-3}, \ldots$ are independent. Furthermore, we assume without loss of generality that in the cyclic case, the server first visits Q_1 after time zero. Define $R_{j,n}^C$ to be the residual preparation time at $Q_{((n+j-1) \bmod N)+1}$ just after the completion of the $(n-1)$-st service in the cyclic case, $n \geq 1$, $j = 1, \ldots, N-2$. Clearly, $R_{N-1,n}^C = B_{n+N-1}$ and $R_{N,n}^C = W_n^C$. It is not hard to see that the process $\{(W_n^C, R_{1,n}^C, R_{2,n}^C, \ldots, R_{N-2,n}^C), n \geq 1\}$ is a discrete-time Markov chain, of which the evolution is given by $W_{n+1}^C = (R_{1,n}^C - W_n^C - A_n)^+$ and $R_{j,n+1}^C = (R_{j+1,n}^C - W_n^C - A_n)^+$ for $j = 1, 2, \ldots, N-2$.

10.3 Analysis of the cyclic waiting-time distribution

In this section, we study the waiting-time distribution of the server in the cyclic model. First, we investigate the existence of a unique limiting waiting-time distribution in Section 10.3.1. Then, we study the tail behaviour of the stationary waiting time in Section 10.3.2 for several classes of preparation time distributions. Finally, Section 10.3.3 shows how to compute the distribution of W_n^C for any $n \geq 1$ under the assumption of exponential preparation times. The analysis presented in this section can conceptually be extended easily to allow for phase-type preparation times.

10.3.1 Existence of a limiting waiting-time distribution

We will argue in this section that a unique limiting waiting-time distribution exists, under the natural assumption that $\mathbb{P}(X \leq 0) > 0$. Note that the stochastic process $\{W_n^C, n \geq 1\}$ is an aperiodic (possibly delayed) regenerative process with regeneration times $\{n : W_n^C = W_{n-1}^C = \cdots = W_{n-2N+4}^C = 0\}$. Colloquially speaking, this is due to the fact that the server's waiting time is independent of past waiting times in case the server did not have to wait in the past two polling cycles. Let j be any regeneration time after $t = 2N-4$. Furthermore, let $\tau = \min\{n : n > 0, W_j^C = W_{j-1}^C = \cdots = W_{j-2N+4}^C = W_{j+n}^C = W_{j+n-1}^C = \cdots = W_{j+n-2N+4}^C = 0\}$, so that τ can be interpreted as the time between two regeneration moments.

We will now show that $\mathbb{E}[\tau]$ is finite, which implies by standard theory on regenerative processes that the limiting distribution of the waiting time exists and that the waiting-time process converges to it (see e.g. [19, Corollary VI.1.5 and Theorem VII.3.6]). To this end, observe that for any $n \geq 2N-3$,

$$\mathbb{P}(\tau > n) = \mathbb{P}\left(\bigcap_{i=j+1}^{j+n} \left\{\sum_{k=0}^{2N-4} W_{i-k}^C > 0\right\}\right) \leq \mathbb{P}\left(\bigcap_{i=j+2N-3}^{j+n} \left\{\sum_{k=0}^{2N-4} W_{i-k}^C > 0\right\}\right).$$

Due to (10.2) and the fact that waiting times are non-negative, X_n is stochastically not smaller than W_n^C. In other words, we have that

$$\mathbb{P}(W_n^C > 0 \mid W_{n-1}^C, W_{n-2}^C, \ldots) \leq \mathbb{P}(X_n > 0 \mid W_{n-1}^C, W_{n-2}^C, \ldots).$$

We also obviously have that $\mathbb{P}(X_n > 0 \mid W_{n-k}^C = 0) \leq \mathbb{P}(X_n > 0)$ for any $k \in \{1, 2, \ldots\}$. As a result, we have for any $n \geq 2N-3$ that

$$\begin{aligned}\mathbb{P}(\tau > n) &\leq \mathbb{P}\left(\bigcap_{i=j+2N-3}^{j+n} \left\{\sum_{k=0}^{2N-4} X_{i-k} > 0\right\}\right) \leq \mathbb{P}\left(\bigcap_{i=1}^{\lfloor \frac{n}{2N-3} \rfloor} \left\{\sum_{k=0}^{2N-4} X_{j+i(2N-3)-k} > 0\right\}\right) \\ &= \mathbb{P}\left(\sum_{k=0}^{2N-4} X_{j+2N-3-k} > 0\right)^{\lfloor \frac{n}{2N-3} \rfloor} < \mathbb{P}\left(\sum_{k=1}^{2N-3} X_{j+k} > 0\right)^{\frac{n}{2N-3}-1}, \end{aligned} \tag{10.3}$$

where the equality follows from the fact that the process $\{X_n, n \geq 0\}$ exhibits no autocorrelation for lag $N-1$ or more. The last inequality holds since $\sum_{k=0}^{2N-4} X_{j+2N-3-k} = \sum_{k=1}^{2N-3} X_{j+k}$ and $\lfloor \frac{n}{2N-3} \rfloor > \frac{n}{2N-3} - 1$. Additionally, we have that

$$\mathbb{P}\left(\sum_{k=1}^{2N-3} X_{j+k} > 0\right) \leq 1 - \mathbb{P}\left(\bigcap_{k=1}^{2N-3} \left\{X_{j+k} \leq 0\right\}\right)$$

$$
\begin{aligned}
&= 1 - \mathbb{P}(X_{j+1} \le 0)\mathbb{P}(X_{j+2} \le 0 \mid X_{j+1} \le 0) \\
&\qquad \times \cdots \times \mathbb{P}\left(X_{j+2N-3} \le 0 \mid \bigcap_{k=1}^{2N-4} \left\{X_{j+k} \le 0\right\}\right) \\
&\le 1 - \mathbb{P}(X \le 0)^{2N-3}. \qquad (10.4)
\end{aligned}
$$

The last inequality holds since the process $\{X_n, n \ge 0\}$ exhibits positive autocorrelation with a lag up to $N-2$, but no autocorrelation for lag $N-1$ or more. Thus, we have that $\mathrm{Cov}[\mathbb{1}_{\{X_{n+k} \le 0\}}, \mathbb{1}_{\{X_n \le 0\}}] \ge 0$ for any $n > N-1$ and $0 < k \le N-2$, so that $\mathbb{P}(X_{n+k} \le 0 \mid X_n \le 0) \ge \mathbb{P}(X \le 0)$. For $k > N-2$, however, we have that $\mathbb{P}(X_{n+k} \le 0 \mid X_n \le 0) = \mathbb{P}(X \le 0)$. Finally, from (10.3), we infer that

$$
\begin{aligned}
\mathbb{E}[\tau] &= \sum_{n=0}^{2N-4} \mathbb{P}(\tau > n) + \sum_{n=2N-3}^{\infty} \mathbb{P}(\tau > n) \le 2N-3 + \sum_{n=0}^{\infty} \mathbb{P}\left(\sum_{k=1}^{2N-3} X_{j+k} > 0\right)^{\frac{n}{2N-3}-1} \\
&\le 2N-3 + \sum_{n=0}^{\infty} (1 - \mathbb{P}(X \le 0)^{2N-3})^{\frac{n}{2N-3}-1} \\
&= 2N-3 + \frac{1}{1-\mathbb{P}(X \le 0)^{2N-3}} \frac{1}{1-(1-\mathbb{P}(X \le 0)^{2N-3})^{\frac{1}{2N-3}}} \\
&< \infty,
\end{aligned}
$$

where the second inequality follows from (10.4). The last inequality holds true under the assumption that $\mathbb{P}(X \le 0) \in (0,1)$. Observe that in the trivial case of $\mathbb{P}(X \le 0) = 1$, the server never waits, resulting in zero waiting times. Therefore, we conclude that a unique limiting distribution exists for the waiting time when $\mathbb{P}(X \le 0) > 0$. The existence of such a distribution in the theoretical case $\mathbb{P}(X < 0) = 0$ is proved in [265, Section 2.2] for $N = 2$, but this result seems hard to extend to a general value of N.

10.3.2 Tail behaviour

We now study the tail behaviour of W^C, the stationary waiting time. For two classes of preparation time distributions, we derive the asymptotic behaviour of the probability that the waiting time W^C exceeds some large value x. The tail behaviour may be useful when, for example, the distribution of W^C cannot be computed exactly or when knowledge on the full distribution of W^C is not necessary. In the remainder of this section, we write $f \sim g$ for two functions $f(x)$ and $g(x)$ when $\lim_{x\to\infty} f(x)/g(x) = 1$. We also require the notion of regularly varying and rapidly varying functions.

A measurable function $f : (0,\infty) \to (0,\infty)$ is called *regularly varying* of a finite index κ if

$$
\lim_{x\to\infty} \frac{f(lx)}{f(x)} = l^{\kappa}
$$

for any $l > 0$. Observe that this definition demands that the index κ is finite. The definition can be extended to include cases for which κ is not finite, leading to the notion of rapid variation. A measurable function $f : (0,\infty) \to (0,\infty)$ is called *rapidly varying* of

index $-\infty$ if it satisfies

$$\lim_{x\to\infty}\frac{f(lx)}{f(x)}=\begin{cases}0 & \text{if } l>1,\\ 1 & \text{if } l=1,\\ \infty & \text{otherwise.}\end{cases}$$

A comprehensive account of the theory and applications of regular variation is given in [38]. By convention, we will call a random variable regularly varying or rapidly varying if its complementary cumulative distribution function has the corresponding property.

We start with the class of preparation time distributions that satisfies

$$\lim_{x\to\infty}\frac{\mathbb{P}(B>x+y)}{\mathbb{P}(B>x)}=e^{-\kappa y}$$

for some finite constant $\kappa\geq 0$, or equivalently,

$$\lim_{x\to\infty}\frac{\mathbb{P}(e^B>e^x e^y)}{\mathbb{P}(e^B>e^x)}=(e^y)^{-\kappa}.$$

Thus, we regard the class of distributions of B for which e^B is a regularly varying random variable with index $-\kappa\leq 0$. For $\kappa=0$, this means that the random variable B is long-tailed (i.e. $\lim_{x\to\infty}\mathbb{P}(B>x+y|B>x)=1$ for all $y>0$) and thus also heavy-tailed (i.e. $\lim_{x\to\infty}e^{\lambda x}\mathbb{P}(B>x)=\infty$ for all $\lambda>0$). If $\kappa>0$, then B is light-tailed, but not lighter than the tail of an exponential distribution.

In order to study the tail behaviour of W^C for this class of preparation time distributions, we will use the following proposition obtained in [64, Corollary 3.6].

PROPOSITION 10.3.1. *If $Y>0$ is a regularly varying random variable with index $-\kappa$, $\kappa\geq 0$, and $Z>0$ is a random variable independent of Y satisfying $\mathbb{E}[Z^{\kappa+\epsilon}]<\infty$ for some $\epsilon>0$, then YZ is also regularly varying with index $-\kappa$. In particular, we have that*

$$\mathbb{P}(YZ>x)\sim\mathbb{E}[Z^{\kappa}]\mathbb{P}(Y>x).$$

Now, let $\overline{Y}=B-A$, and let $\overline{Z}$ be a random variable with a distribution equal to the limiting distribution of $W_n^C+\sum_{i=n-N+2}^{n-1}(A_i+W_i^C)$ as $n\to\infty$ under the conditions of Section 10.3.1. Then, we have due to the recursion in (10.1) that $W^C\stackrel{d}{=}\overline{Y}-\overline{Z}$. The following theorem states that the tail of W behaves asymptotically as the tail of B or the tail of $\overline{Y}$, multiplied by a constant.

THEOREM 10.3.2. *Let e^B be regularly varying with index $-\kappa$, $\kappa>0$. Then, we have for the tail of W^C that*

$$\mathbb{P}(W^C>x)\sim\mathbb{E}[e^{-\kappa(A+\overline{Z})}]\mathbb{P}(B>x) \text{ and } \mathbb{P}(W^C>x)\sim\mathbb{E}[e^{-\kappa\overline{Z}}]\mathbb{P}(\overline{Y}>x).$$

PROOF. We have from (10.1) that $\mathbb{P}(W^C>x)=\mathbb{P}(B-A-\overline{Z}>x)$, or equivalently, that $\mathbb{P}(e^{W^C}>e^x)=\mathbb{P}(e^B e^{-(A+\overline{Z})}>e^x)$. Note that $e^{-(A+\overline{Z})}$ is a positive random variable, which for any $\epsilon>0$ satisfies

$$\mathbb{E}[e^{-(\kappa+\epsilon)(A+\overline{Z})}]\leq 1<\infty,$$

as $A+\overline{Z}$ cannot take negative values. Therefore, we obtain by applying Proposition 10.3.1 with $Y = e^B$ and $Z = e^{-(A+\overline{Z})}$ that

$$\mathbb{P}(e^{W^C} > e^x) \sim \mathbb{E}[e^{-\kappa(A+\overline{Z})}]\mathbb{P}(e^B > e^x) = \mathbb{E}[e^{-\kappa(A+\overline{Z})}]\mathbb{P}(B > x).$$

For the second part of the theorem, note that $\mathbb{E}[e^{-(\kappa+\epsilon)A}] \leq 1 < \infty$ for any $\epsilon > 0$ as A only takes non-negative values. Therefore, since e^B is regularly varying with index $-\kappa$, $e^{\overline{Y}}$ is too by Proposition 10.3.1. The expression for the tail of W^C in terms of the tail of $\overline{Y}$ now follows from an analysis similar to the one above using Proposition 10.3.1 with $Y = e^{\overline{Y}}$ and $Z = e^{-\overline{Z}}$. □

An example of a random variable B that satisfies the conditions of this theorem is the one asymptotically having the tail distribution $\mathbb{P}(B > x) \sim c_0 x^{c_1} \mathrm{e}^{-c_2 x}$ for some real-valued constants c_i, $i = 0, 1, 2$, where $c_0, c_2 > 0$.

We now consider the class of preparation time distributions for which e^B is rapidly varying with index $-\infty$, that is

$$\lim_{x\to\infty} \frac{\mathbb{P}(e^B > e^x e^y)}{\mathbb{P}(e^B > e^x)} = \lim_{x\to\infty} \frac{\mathbb{P}(B > x+y)}{\mathbb{P}(B > x)} = \begin{cases} 0 & \text{if } y > 0, \\ 1 & \text{if } y = 0, \\ \infty & \text{if } y < 0. \end{cases}$$

This is equivalent to letting the index κ that was given previously go to infinity. For the random variable B, this means that it is extremely light-tailed. As an example, one can think of a distribution for which the complementary cumulative distribution function is given by $\mathbb{P}(B > x) = e^{-x^p}$, where $p > 1$.

For this class of preparation time distributions, we derive the asymptotic behaviour of the tail of W^C, under the assumption that $\mathbb{P}(\overline{Z} = 0) > 0$. Thus, we assume among other things that the distribution of A has an atom at zero. The following theorem states that, as before, the tail of W^C then behaves asymptotically as the tail of $\overline{Y}$ multiplied by a constant. A similar result under more general assumptions on the distribution of A and B seems hard to obtain unless $N = 2$ (cf. [265]).

THEOREM 10.3.3. *Let e^B be rapidly varying with index $-\infty$. If $\mathbb{P}(\overline{Z} = 0) > 0$, the tail of W^C satisfies*

$$\mathbb{P}(W^C > x) \sim \mathbb{P}(\overline{Y} > x)\mathbb{P}(\overline{Z} = 0).$$

PROOF. Note that according to (10.1),

$$\begin{aligned} \mathbb{P}(W^C > x) &= \lim_{n\to\infty} \mathbb{P}\left(B_n - \sum_{i=n-N+2}^{n} A_i - \sum_{i=n-N+2}^{n} W_i^C > x \right) && (10.5) \\ &= \mathbb{P}(\overline{Y} - \overline{Z} > x) \\ &= \mathbb{P}(\overline{Y} > x)\mathbb{P}(\overline{Z} = 0) + \mathbb{P}(\overline{Y} - \overline{Z} > x \mid 0 < \overline{Z} < \epsilon)\mathbb{P}(0 < \overline{Z} < \epsilon) \\ &\quad + \mathbb{P}(\overline{Y} - \overline{Z} > x \mid \overline{Z} \geq \epsilon)\mathbb{P}(\overline{Z} \geq \epsilon) && (10.6) \end{aligned}$$

for some $\epsilon > 0$. Since the last two terms of the right-hand side of (10.6) are non-negative, we conclude immediately that

$$\liminf_{x\to\infty} \frac{\mathbb{P}(W^C > x)}{\mathbb{P}(\overline{Y} > x)\mathbb{P}(\overline{Z} = 0)} \geq 1. \tag{10.7}$$

Concerning the upper limit, observe that $\mathbb{P}(\overline{Y}-\overline{Z}>x \mid 0<\overline{Z}<\epsilon) \leq \mathbb{P}(\overline{Y}>x)$ and that $\mathbb{P}(\overline{Y}-\overline{Z}>x \mid \overline{Z}\geq\epsilon) \leq \mathbb{P}(\overline{Y}>x+\epsilon)$. As e^B is rapidly varying, $e^{\overline{Y}}$ is too (see e.g. [265, Lemma 1]). Therefore, we have for $\epsilon>0$ that

$$\lim_{x\to\infty} \frac{\mathbb{P}(\overline{Y}>x+\epsilon)}{\mathbb{P}(\overline{Y}>x)} = 0.$$

Combining the above arguments, we obtain from (10.6) that

$$\limsup_{x\to\infty} \frac{\mathbb{P}(W^C>x)}{\mathbb{P}(\overline{Y}>x)\mathbb{P}(\overline{Z}=0)} \leq 1+\frac{\mathbb{P}(0<\overline{Z}<\epsilon)}{\mathbb{P}(\overline{Z}=0)}. \tag{10.8}$$

By taking the limit $\epsilon\to 0$, we therefore have that

$$\limsup_{x\to\infty} \frac{\mathbb{P}(W^C>x)}{\mathbb{P}(\overline{Y}>x)\mathbb{P}(\overline{Z}=0)} = 1,$$

since the inequalities in $\mathbb{P}(0<\overline{Z}<\epsilon)$ are strict, $\mathbb{P}(\overline{Z}=0)$ is positive and the left-hand side of (10.8) does not depend on ϵ. Combining (10.7) with this expression now leads to the theorem. □

10.3.3 Transient analysis

In this section, we assume that preparation times are exponentially distributed with rate μ. Note that the analysis can extend to phase-type preparation times, but at the cost of more cumbersome expressions. Furthermore, little insight is added by such an extension. We first show that the waiting time has an atom at zero and, provided that it is positive, is also exponentially distributed with rate μ. We then calculate the atom at zero by computing the transition matrix of the underlying discrete-time Markov chain. We show that the matrix has a nice structure that can be exploited for numerical computations.

10.3.3.1 The behaviour of W^C_{n+1}

We show that the waiting time, given that it is positive, is exponentially (μ) distributed. For $n\geq N-1$, we have that

$$\begin{aligned}
&\mathbb{P}\left(W^C_{n+1}>x \mid W^C_n=w_n,\ldots,W^C_{n-N+2}=w_{n-N+2}\right)\\
&=\mathbb{P}\left(B_{n+1}>\sum_{i=n-N+2}^{n} A_i+\sum_{i=n-N+2}^{n} w_i+x\right)\\
&=\int_{y_{n-N+2}=0}^{\infty}\cdots\int_{y_n=0}^{\infty} e^{-\mu(\sum_{i=n-N+2}^{n}(y_i+w_i)+x)}dF_{A_n}(y_n)\ldots dF_{A_{n-N+2}}(y_{n-N+2})\\
&=(\widetilde{A}(\mu))^{N-1}e^{-\mu(\sum_{i=n-N+2}^{N} w_i+x)},
\end{aligned} \tag{10.9}$$

where we defined $\widetilde{A}(\mu) = \mathbb{E}[e^{-\mu A}]$. From this equation, we conclude that

$$\begin{aligned}
&\mathbb{P}(W_{n+1}^C > x \mid W_{n+1}^C > 0, W_n^C = w_n, \ldots, W_{n-N+2}^C = w_{n-N+2}) \\
&= \frac{\mathbb{P}(W_{n+1}^C > x \mid W_n^C = w_n, \ldots, W_{n-N+2}^C = w_{n-N+2})}{\mathbb{P}(W_{n+1}^C > 0 \mid W_n^C = w_n, \ldots, W_{n-N+2}^C = w_{n-N+2})} \\
&= \frac{(\widetilde{A}(\mu))^{N-1} e^{-\mu(\sum_{i=n-N+2}^{n} w_i + x)}}{(\widetilde{A}(\mu))^{N-1} e^{-\mu(\sum_{i=n-N+2}^{n} w_i)}} = \mathrm{e}^{-\mu x},
\end{aligned}$$

meaning that W_{n+1}^C, provided that it is positive, is not affected by the previous $N-1$ waiting times. A direct conclusion is that $\mathbb{P}(W_{n+1}^C > x \mid W_{n+1}^C > 0) = e^{-\mu x}$, so that

$$\begin{aligned}
&\mathbb{P}(W_{n+1}^C > x) \\
&= \mathbb{P}(W_{n+1}^C > x \mid W_{n+1}^C > 0)\mathbb{P}(W_{n+1}^C > 0) + \mathbb{P}(W_{n+1}^C > x \mid W_{n+1}^C = 0)\mathbb{P}(W_{n+1}^C = 0) \\
&= e^{-\mu x}\mathbb{P}(W_{n+1}^C > 0). \qquad (10.10)
\end{aligned}$$

That is, the distribution of W_n^C is a mixture of a mass at zero and the exponential distribution with rate μ, in case $n \geq N-1$. The same result for $1 \leq n < N-1$ follows by performing a similar analysis. The argument can also be applied to W^C, the limit of W_n^C as $n \to \infty$, so that $\mathbb{P}(W^C > x) = e^{-\mu x}\mathbb{P}(W^C > 0)$. We now calculate $\mathbb{P}(W_{n+1}^C > 0)$ for all n, and $\mathbb{P}(W^C > 0)$. To this end, we will define a discrete-time Markov chain and calculate its one-step transition probability matrix.

10.3.3.2 Construction of a discrete-time Markov chain

Recall that the process $\{(W_n^C, R_{1,n}^C, R_{2,n}^C, \ldots, R_{N-2,n}^C), n \geq 1\}$ is a discrete-time Markov chain. We have just shown that W_n^C, provided that it is positive, is distributed according to B irrespective of the previous waiting times when B follows an exponential distribution. It is also trivial to see that a residual preparation time $R_{j,N}^C$, given that it is positive, has the same distribution as B, because of the memoryless property of the exponential distribution. Due to these observations, the process $\{(F_n^C, G_{1,n}^C, \ldots, G_{N-2,n}^C), n \geq 1\}$ is a discrete-time Markov chain on the state space $\mathscr{S}^C = \{0,1\}^{N-1}$, where $F_n^C = \mathbb{1}_{\{W_n^C > 0\}}$ and $G_{j,n}^C = \mathbb{1}_{\{R_{j,n}^C > 0\}}$. A state $\boldsymbol{i} = (i_1, \ldots, i_{N-1}) \in \mathscr{S}^C$ describes the residual preparation time at each station (positive or zero) at the start of the n-th waiting time of the server (including zero waiting times). The only station that does not appear in this description is the station the server has just served before this instant, since the residual preparation time there is always larger than zero (or, in other words, $G_{N-1,n}^C = 1$ for all n).

Before we derive the one-step transition probabilities of this discrete-time Markov chain, we first observe that the chain, provided that it is in state $\boldsymbol{i} \in \mathscr{S}^C$, may not be able to transition directly to any state $\boldsymbol{j} \in \mathscr{S}^C$. This is a result of the fact that a preparation phase that is already completed when transitioning to state $\boldsymbol{i}$, obviously remains completed until after the following transition, unless its corresponding service station is served in between the two transitions. In that case, a new preparation phase starts at the next transition. In order words, the chain can only move from a state $\boldsymbol{i}$ to a state $\boldsymbol{j}$ when $j_{k-1} = 0$ for each $k \in \{2, \ldots, N-1\}$ for which $i_k = 0$. Therefore, we define the set $T(\boldsymbol{i}) = \{\boldsymbol{j} : j_{k-1} \leq i_k \ \forall k \in \{2, \ldots, N-1\}\}$ to be the set of possible states the chain can transition to after a visit to state $\boldsymbol{i}$. For any state $\boldsymbol{i}$, we also define $k_{\boldsymbol{i}} = \sum_{r=1}^{N-1} i_r$ to be

the number of preparation phases that are in progress just before the system moves to state $\boldsymbol{i}$. Finally, we define $d_{\boldsymbol{i},\boldsymbol{j}} = k_{\boldsymbol{i}} - k_{\boldsymbol{j}}$ to be the difference between these numbers corresponding to states $\boldsymbol{i}$ and $\boldsymbol{j}$. Using these definitions, we can now obtain the one-step transition probabilities $P_{\boldsymbol{i},\boldsymbol{j}}$ from any state $\boldsymbol{i} \in \mathscr{S}^C$ to any state $\boldsymbol{j} \in \mathscr{S}^C$. These results are summarised in the following proposition.

PROPOSITION 10.3.4. *The one-step transition probabilities of the discrete-time Markov chain* $\{(F_n^C, G_{1,n}^C, \ldots, G_{N-2,n}^C), n \geq 1\}$ *are given by*

$$P_{\boldsymbol{i},\boldsymbol{j}} = \begin{cases} \sum_{l=0}^{d_{i,j}+1} \binom{d_{i,j}+1}{l} (-1)^l \widetilde{A}((k_{\boldsymbol{j}}+l)\mu) & \text{if } i_1 = 0 \text{ and } \boldsymbol{j} \in T(\boldsymbol{i}), \\ \sum_{l=0}^{d_{i,j}} \binom{d_{i,j}}{l} (-1)^l \frac{\widetilde{A}((k_{\boldsymbol{j}}+l)\mu)}{k_{\boldsymbol{j}}+l+1} & \text{if } i_1 = 1 \text{ and } \boldsymbol{j} \in T(\boldsymbol{i}), \\ 0 & \text{otherwise} \end{cases}$$

for any $\boldsymbol{i}, \boldsymbol{j} \in \mathscr{S}^C$.

PROOF. When $i_1 = 0$ and $\boldsymbol{j} \in T(\boldsymbol{i})$, a service phase starts immediately when the discrete-time Markov chain enters state $\boldsymbol{i}$. Therefore, the time between the transition to state $\boldsymbol{i}$ and the next transition to state $\boldsymbol{j}$ amounts exactly to the duration of this service phase. As the transition to state $\boldsymbol{i}$ marks the start of a new preparation phase at the service station served just before this transition, the number of preparation phases in progress just after this transition equals $k_{\boldsymbol{i}} + 1$. If the chain then transitions to $\boldsymbol{j}$, it means that exactly $k_{\boldsymbol{j}}$ of these preparation phases should still be in progress after the transition to state $\boldsymbol{j}$. The other $(k_{\boldsymbol{i}} + 1) - k_{\boldsymbol{j}} = d_{\boldsymbol{i},\boldsymbol{j}} + 1$ preparation phases must finish over the course of a service time A. Therefore, we have in this case that

$$P_{\boldsymbol{i},\boldsymbol{j}} = \int_{y=0}^{\infty} (1 - e^{-\mu y})^{d_{i,j}+1} e^{-k_{\boldsymbol{j}}\mu y} dF_A(y) = \sum_{l=0}^{d_{i,j}+1} \binom{d_{i,j}+1}{l} (-1)^l \widetilde{A}((k_{\boldsymbol{j}}+l)\mu).$$

When $i_1 = 1$ and $\boldsymbol{j} \in T(\boldsymbol{i})$, the time until the transition to state $\boldsymbol{j}$ does not only consist of a service time A, but also of some waiting time needed for the preparation phase at the server's location to finish. We have seen that the distribution of this waiting time equals that of B, independently of other waiting times (cf. (10.10)). Of the $k_{\boldsymbol{i}} + 1$ preparation phases just after the transition to state $\boldsymbol{i}$, the preparation phase at the server's location finishes at any rate before the next transition. Consequently, for the chain to transition from state $\boldsymbol{i}$ to state $\boldsymbol{j}$, exactly $k_{\boldsymbol{j}}$ of the remaining $k_{\boldsymbol{i}}$ preparation phases must still be in progress after the transition to state $\boldsymbol{j}$, and the other $k_{\boldsymbol{i}} - k_{\boldsymbol{j}} = d_{\boldsymbol{i},\boldsymbol{j}}$ should not. Thus, for this case, we have that

$$\begin{aligned} P_{\boldsymbol{i},\boldsymbol{j}} &= \int_{x=0}^{\infty} \int_{y=0}^{\infty} (1 - e^{-\mu(x+y)})^{d_{i,j}} e^{-k_{\boldsymbol{j}}\mu(x+y)} \mu e^{-\mu x} dF_A(y) dx \\ &= \sum_{l=0}^{d_{i,j}} \binom{d_{i,j}}{l} (-1)^l \frac{\widetilde{A}((k_{\boldsymbol{j}}+l)\mu)}{k_{\boldsymbol{j}}+l+1}. \end{aligned}$$

Finally, it is obvious by the definition of $T(\boldsymbol{i})$ that $P_{\boldsymbol{i},\boldsymbol{j}} = 0$ if $\boldsymbol{j} \notin T(\boldsymbol{i})$. This completes the derivation of the one-step transition probability matrix. □

Now that the one-step transition probabilities are derived, the one-step transition probability matrix $P = (P_{\boldsymbol{i},\boldsymbol{j}})_{\boldsymbol{i},\boldsymbol{j}\in\mathscr{S}^C}$ can be constructed, e.g. by arranging all states in lexicographic order. Using this matrix, one can compute the unknown $\mathbb{P}(W_n^C > 0)$ needed to obtain the transient distribution of W_n^C for any n (cf. (10.10)) or, in case $n \to \infty$, the stationary distribution of W^C. Assume that the system starts in an arbitrary state $\boldsymbol{k} \in \mathscr{S}^C$. Let $\boldsymbol{e}_{\boldsymbol{k}}$ be the unit vector of which the entry at the index which corresponds to state $\boldsymbol{k}$ equals one (and all other elements equal zero). Then, by standard theory on discrete-time Markov chains, $\mathbb{P}(W_n^C > 0)$ equals the sum of the entries of the vector $\boldsymbol{e}_{\boldsymbol{k}}P^{n-1}$ that, according to the ordering of states chosen, correspond to states for which the first element equals one (i.e. a non-zero waiting time). Likewise, the steady-state probability $\mathbb{P}(W^C > 0)$ can be found by computing the unique vector $\boldsymbol{\pi}$ satisfying $\boldsymbol{\pi} = \boldsymbol{\pi}P$ and $\sum_{i\in\mathscr{S}^C} \pi_i = 1$. The probability $\mathbb{P}(W^C > 0)$ is then again given by the sum of the entries of $\boldsymbol{\pi}$ that correspond to states of which the first element equals one. This concludes the analysis of the size of the probability mass at zero for exponentially distributed preparation times.

REMARK 10.3.1. In this section, we assumed that preparation times are equally distributed at each of the service stations. One might also be interested in the case where the duration of a customer's preparation phase at service station Q_i is exponential with a station-specific rate μ_i. Then, it follows immediately from the analysis leading up to (10.10) that the server's waiting time at Q_i, provided that it is positive, is also exponentially (μ_i) distributed. Furthermore, the size of the mass at zero can still be computed by constructing a discrete-time Markov chain, using the same conceptual methods. However, in this case the position of the server needs to be included in the state space to retain the Markov property, and the residual preparation times in the system are not necessarily identically distributed anymore. Therefore, the expressions will become more cumbersome, providing little additional insight into the behaviour of the system.

REMARK 10.3.2. In this section, we mainly studied the waiting time W^C of the server as a performance measure. Another important performance measure pertaining to the system is the throughput θ^C, i.e. the mean number of customers that finish their service per unit of time. Observe that θ^C is equal to the number of customers N served per cycle over the expected cycle length, which has duration $N(\mathbb{E}[W^C] + \mathbb{E}[A])$. Thus, we have that

$$\theta^C = (\mathbb{E}[W^C] + \mathbb{E}[A])^{-1};$$

see also [188]. Consequently, the results of this section can be readily applied to analyse the throughput of the system, since $\mathbb{E}[A]$ is a known constant. In Section 10.4, we will focus on the impact of the parameter settings on the throughput of the system.

10.4 Insights

In the previous sections, we gave closed-form expressions for exponentially distributed preparation times. Here, we gain general insights into the behaviour of the cyclic model by simulation on a larger range of parameter settings. We vary, among other things, the number of stations and the distributions of the preparation and service times. We focus on the effect of the first two moments of the preparation and service times on the throughput. For their distributions, we choose phase-type distributions based on two-moment-fit approximations commonly used in literature; see e.g. [238, pp. 358–360]. We discuss several interesting conclusions based on the simulation results.

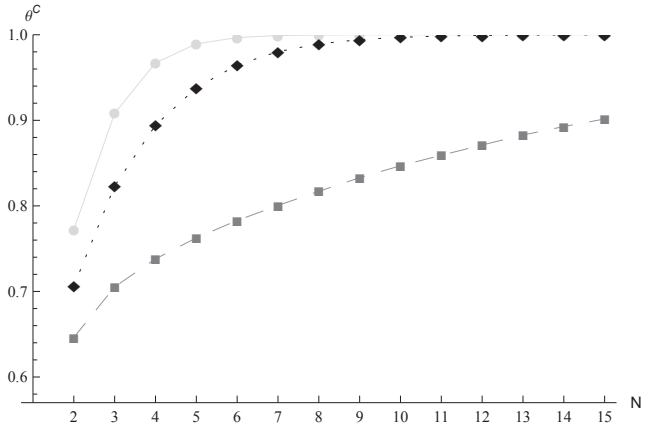

FIGURE 10.1: Throughput as a function of the number of stations for moderately variable preparation and service times (solid), highly variable service times (dotted) and highly variable preparation times (dashed).

Variability of preparation and service times When controlling the system, the variability of the preparation times seems to play a larger role than the variability of the service time. This is because the server's waiting-time process is much more sensitive to the former than to the latter. See e.g. Figure 10.1, where the throughput θ^C is plotted versus the number of queues N. We observe the throughput for various variability settings for both time components. We fix the means at $\mathbb{E}[A] = \mathbb{E}[B] = 1$ and first consider the same phase-type distributions with low variability for both the preparation and service time, i.e. $\mathbb{E}[A^2] = \mathbb{E}[B^2] = 1.5$ (solid curve). We also consider the case with highly variable service times only, i.e. $\mathbb{E}[A^2] = 10, \mathbb{E}[B^2] = 1.5$ (dotted curve) and highly variable preparation times only, i.e. $\mathbb{E}[A^2] = 1.5, \mathbb{E}[B^2] = 10$ (dashed curve). Although the variability of the preparation times and the service times is varied in similar ways, the dotted curve nears the solid curve as N grows larger much faster than the dashed curve. Therefore, predictability of the preparation times seems to be much more important than that of the service times. This can be explained by the fact that as the number of stations tends to infinity, the squared coefficient of variation of the sum of service times in the right-hand side of (10.1) goes to zero, and thus the effect of any variability in the service times is less serious. In other words, in service systems, it is more important that one has a reliable assistant than a reliable server. This holds in particular for large systems. In the warehousing setting as described in Section 1.3.3, this is more or less guaranteed; although the preparation times (i.e. rotation times) depend on the picking strategy followed, they are bounded by the length of the carousel and therefore exhibit small variability. Whether the picker is robotic (small variability) or human, does influence the system, but not as dramatically as the preparation times do.

A similar effect is observed in Figure 10.2, where the mean number of positive waiting times $\mathbb{E}[C^C]$ between two zero waiting times is plotted versus the second moment of the preparation time B (solid curve) or that of the service time A (dashed curve). It is assumed that $N = 4$ and $\mathbb{E}[A] = \mathbb{E}[B] = 1$ throughout for both of these lines. For the first curve, the service times A are taken to be exponentially distributed, while for the second, the

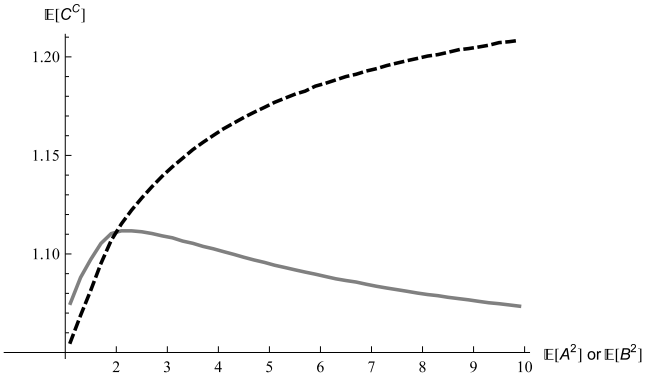

FIGURE 10.2: Mean time between two zero waiting times as a function of $\mathbb{E}[B^2]$ (solid) and $\mathbb{E}[A^2]$ (dashed).

preparation times B are taken to be exponentially distributed. From Figure 10.2, it is apparent that the mean time between two zero waiting times increases (i.e. the frequency of zero waiting times decreases) as the service times become more variable. However, mostly the opposite is observed for the preparation times. Although the expected waiting time increases in the variability of the preparation times by Figure 10.1, apparently the mean time between two zero waiting times now *decreases* anomalously. From this, we conclude that the server's waiting time process behaves more and more erratically as the variability of the preparation times increases and seems to be more resistant against highly variable service times. Again, this effect may be explained by the nature of the waiting time (see (10.1)), which is expressed in terms of one preparation time, but a *sum* of service times. No matter the variability of the individual service times, the sum of these service times behaves more and more deterministically as N increases. In other words, the effect of highly variable service times is mitigated by the fact that the waiting time only depends on a sum of them.

In summary, we can say that an increase in the variability of preparation times, as long as the variability is small, makes the server wait less frequently. This also holds for an increase in the variability in the service times, independent of the degree of variability. However, both scenarios decrease the throughput of the system. Thus, when waiting times occur, they tend to be longer. Simulation results show about a 10% decrease in throughput under common scenarios when ranging the distribution of the preparation time from deterministic to exponential (thus ranging the squared coefficient of variation from zero to one). Nonetheless, in some service systems, this may be an advantage, as it gives the opportunity to perform an additional task (e.g. administration).

Correlations In general, this system has an interesting correlation structure. In Figure 10.3, we plot the stationary autocorrelation coefficient of lag k between two waiting times for exponential preparation and service times with rates 1 and 10, respectively. As we see in Figure 10.3, correlations exhibit a periodic structure, which is natural as it corresponds to a return to the first station. Moreover, as the lag increases, the waiting times become uncorrelated, which is again a natural conclusion. As shown in Section 10.3.1, there exists a unique limiting waiting-time distribution and the system converges to it.

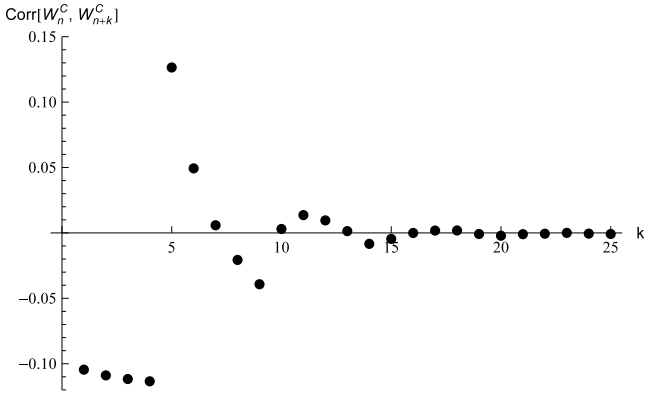

FIGURE 10.3: Stationary autocorrelation coefficients of the waiting times in the cyclic model.

Hence, as time goes to infinity, the system converges to steady state regardless of the initial state. As a result, the correlation coefficient pertaining to lag k goes to zero as $k \to \infty$. Although the convergence to zero correlations is expected, the way this happens is intriguing. One may expect some form of periodicity, but it is not clear why the first cycle looks different than the rest or why correlations should be forming alternatingly convex and concave loops after the first cycle.

Number of stations to be assigned to a server One of the important management decisions to be made is the number of stations to be assigned to a server. For instance, in the warehousing setting as given in Section 1.3.3, the more carousels there are assigned to the picker, the better his utilisation. However, the utilisation of each carousel decreases. We wish to understand this interplay. An important measure to be taken into account is the throughput of the system. Note that the throughput is linearly related to the fraction of time the server is operating, since service is completed at rate $1/\mathbb{E}[A]$ whenever the server is not forced to wait. The number of stations to be assigned to a server in order to reach near-optimal throughput depends very much on the distributions of the preparation time B and the service time A. This effect is observed in Figure 10.1, where we see that for highly variable preparation times (dashed line), the throughput does not converge very fast to the optimal throughput when assigning additional stations to the server. Variability in the service times also influences the system, but the convergence follows more or less the pattern of the case with $\mathbb{E}[A^2] = \mathbb{E}[B^2] = 1.5$.

When all distributions are exponential, it is evident that the only quantity that matters in the determination of the throughput is $r = \mathbb{E}[B]/\mathbb{E}[A]$. In order to determine the optimal number of stations to assign to a server, we plot in Figure 10.4 the throughput θ^C versus the number of stations N for three cases of r, namely for $r = 0.5$ (dashed curve), $r = 1$ (solid curve) and $r = 2.0$ (dotted curve). In all three cases, the underlying distributions are exponential. What we observe is that when $r \leq 1$ (the top two curves), the throughput converges fast and little benefit is added by assigning one more station to the server. This is to be expected, as in this case, the mean service time is not smaller than the mean preparation time. As a result, the server rarely has to wait. In other words, he

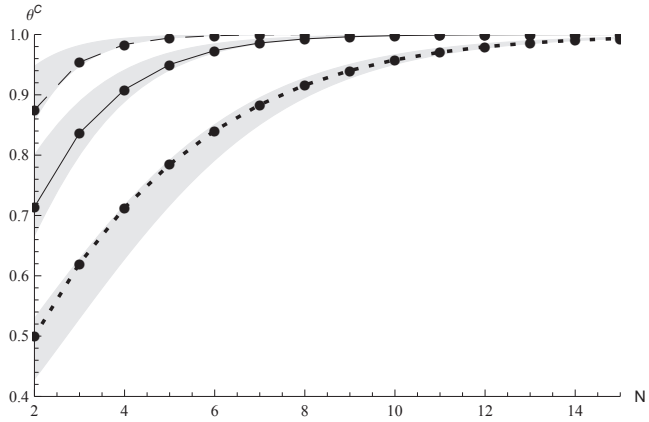

FIGURE 10.4: Throughput as a function of the number of stations for small (dashed), moderate (solid) and large preparation times (dotted).

works at almost full capacity, and thus convergence to the maximum service rate is fast. However, when $r > 1$, the convergence is very slow. We conclude that the shape of the distribution plays a role, but in general for $r \leq 1$ and low variability in preparation times, only few stations per server (about 5 or 6) are needed to already come close to the maximum throughput.

A rough estimate In Figure 10.4, we also plot a rough upper bound and an analytic lower bound of the throughput that we derive as follows. Recall that the throughput θ^C satisfies

$$\theta^C = (\mathbb{E}[W^C] + \mathbb{E}[A])^{-1}.$$

An approximation $\hat{\theta}_N^C$ of θ^C can be produced by replacing $\mathbb{E}[W^C]$ with the mean residual preparation time multiplied by a rough estimate that the server has to wait, i.e. $\mathbb{P}(B > A_1 + \cdots + A_{N-1})$. Then, for exponentially (μ) distributed preparation times, this leads to

$$\hat{\theta}_N^C = \left(\frac{(\widetilde{A}(\mu))^{N-1}}{\mu} + \mathbb{E}[A] \right)^{-1}.$$

We observe that this expression is a lower bound of the throughput, since the actual (stationary) probability a server has to wait equals $\lim_{n\to\infty} \mathbb{P}(B > A_{n-N+2} + \cdots + A_n + W_{n-N+2} + \cdots + W_n)$ and is thus smaller. We also observe empirically that $\hat{\theta}_{N+1}^C$ provides an upper bound for the throughput for the model with N stations in the scenarios we examined. The analytic lower bound becomes tighter as r increases, while the empirical upper bound provides a better estimate for small values of r. As a result, the system's designer can have a quick, easy and accurate bound on the throughput for all parameter settings.

11

COMPARISON WITH A DYNAMIC MODEL

We now drop the cyclic assumption of the previous chapter and investigate the dynamic model variation where the server always serves the customer with the earliest completed preparation phase. As in the previous chapter, we analyse the waiting-time distribution of the server by constructing an appropriate discrete-time Markov chain. Furthermore, we show that the mean waiting time under this dynamic allocation never exceeds that of the cyclic model, but that the waiting-time distributions corresponding to both models are not necessarily stochastically ordered. Finally, we provide extensive numerical results for the dynamic model, which reveal the effects of the model parameters on the performance of the system. We compare these effects to those of the cyclic model as obtained in the previous chapter.

11.1 Introduction

In this chapter, we study the question of how the waiting-time distribution of the server is affected when we drop the assumption that the server is forced to visit the stations cyclically. After a service, in an effort to reduce his overall waiting time, the server will instead visit the service station corresponding to the customer who completes or has had its preparation completed earlier than all of the other customers that are first in line at the other service stations. It is thus also possible for the server to serve two customers in a row at the same service station in case all other service stations still have a preparation phase in progress when the preparation phase following the service of the first customer completes. In short, the order in which stations are served now becomes dynamic. The removal of the cyclic assumption has a significant impact on the analysis, since the waiting time in the new dynamic model does not satisfy a Lindley-type equation anymore. So far, results comparing the two models were already derived in [266] for the special case of two service stations, but these results generally either do not hold for a larger number of service stations or their derivation is not trivially extended to a general number of service points. In this chapter, we explicitly consider model instances with more than two service stations.

As mentioned in Section 1.3.3, the dynamic model which arises after removal of the cyclic assumption turns out to be equivalent to the extended machine repair problem described in Section 1.3.1. In particular, the service stations then represent the machines,

and the server coincides with the repairman. Furthermore, the preparation and the service phases coincide with the breakdown and repair times of the server. Thus, our study of the waiting times of the server is equivalent to that of the idle times of the repairman between the end of a repair of one machine and the breakdown of the next machine, given that all machines are working in between. These idle times have not been studied extensively in the classical literature on the machine repair problem, perhaps because the operating time of the machine is usually more valuable than the utilisation of the repairman. In our setting, however, we are concerned with the idle times of the repairman.

We use the same notation as in Chapter 10. However, when dealing with the dynamic model, we refer to the n-th waiting time of the server as W_n^D so as to clearly distinguish between the waiting-time distributions of the cyclic and the dynamic model. As the number of station visits between two visits of the same station is now stochastic, there is no simple equivalent of (10.1) available for the waiting times $\{W_n^D, n \geq 0\}$ of the server in the dynamic case. When defining $R_{j,n}^D$ to be the residual preparation time at Q_j just after the $(n-1)$-st service, the process $\{(R_{1,n}^D, \ldots, R_{N,n}^D), n \geq 1\}$ also forms a discrete-time Markov chain. Evidently, we have that $W_n^D = \min_{j\in\{1,\ldots,N\}}\{R_{j,n}^D\}$. Furthermore, we have that $R_{j,n}^D$ is an independent copy of B if the $(n-1)$-st customer was served at Q_j. Otherwise, we have that $R_{j,n+1}^D = (R_{j,n}^D - W_n^D - A_n)^+$.

As before, we study the limiting waiting-time distribution of the server by constructing an appropriate discrete-time Markov chain in Section 11.2. We then compare this distribution with the cyclic case. Although we will see in Section 11.3 that there is no stochastic ordering in the distributions in general, it is proved that the mean of the waiting time under the dynamic allocation policy never exceeds the mean waiting time incurred if the server were to visit the service stations in a cyclic order. By means of a numerical study, we also comment in Section 11.4 on how the insights gained in Section 10.4 change when exchanging the cyclic policy for the dynamic policy. In particular, it turns out that although the variability of the preparation times has a big influence on the system in the cyclic case, the waiting time of the server is almost insensitive to this variability in the dynamic case. In the previous chapter, we also saw that having a very small variability of the preparation times makes the server wait less frequently, but decreases the throughput of the system. However, this does not occur for the dynamic model either. Furthermore, when dropping the cyclic assumption, it turns out that fewer service stations per server are required to guarantee a high utilisation rate of the server, since the expected waiting time of the server drops dramatically. Finally, the autocorrelation structure of the waiting times for the dynamic model turns out to behave very differently from that of the cyclic model.

11.2 Analysis of the dynamic waiting-time distribution

As in the cyclic case, the waiting-time distribution of the server can be analysed using a Markov chain approach when assuming phase-type preparation times. For exponentially (μ) distributed preparation times, the waiting-time distribution is obtained as follows. Evidently, a non-zero waiting time occurs in the system only if just after the end of a service, there is a preparation phase in progress at every service station. The waiting time then lasts until one of these N preparation times finishes. Due to the memoryless property of the exponential distribution, the waiting time, provided that it is positive, is

thus exponentially $(N\mu)$ distributed:

$$\mathbb{P}(W_n^D > x) = e^{-N\mu x}\mathbb{P}(W_n^D > 0).$$

The analysis thus again boils down to the computation of the size of the atom at zero. To this end, we again formulate a discrete-time Markov chain similarly to Section 10.3.3. Let Z_n^D be the number of preparation phases in progress in the complete system just after the service of the n-th customer. Then, again due to the memoryless property of the exponential distribution, the process $\{Z_n^D, n \geq 0\}$ constitutes a discrete-time Markov chain on the state space $\mathscr{S}^D = \{1, \ldots, N\}$. Observe that zero is not included in the state space, as the end of a service always marks the start of a preparation phase. The one-step transition probability from state i to state j is then given by

$$P_{i,j} = \begin{cases} \binom{i}{j-1}\sum_{k=0}^{i-j+1}\binom{i-j+1}{k}(-1)^k\widetilde{A}((k+j-1)\mu) & \text{if } i \in \mathscr{S}^D\backslash\{N\},\ j \in \{1,\ldots,i+1\}, \\ \binom{N-1}{j-1}\sum_{k=0}^{N-j}\binom{N-j}{k}(-1)^k\widetilde{A}((k+j-1)\mu) & \text{if } i = N,\ j \in \mathscr{S}^D, \\ 0 & \text{otherwise.} \end{cases}$$

The expression for $i \in \{1, \ldots, N-1\}$ and $j \in \{1, \ldots, i+1\}$ follows by noting that in that case $i-j+1$ preparation phases have been completed during the service time that marks the transition, and $j-1$ preparation phases have not. The distribution of the number of phases completed during this service time A is obviously binomially distributed with parameters $i-1$ and $1-e^{-\mu A}$. Therefore, we have that

$$\begin{aligned} P_{i,j} &= \int_{x=0}^{\infty}\binom{i}{j-1}(1-e^{-\mu x})^{i-j+1}(e^{-\mu x})^{j-1}dF_A(x) \\ &= \binom{i}{j-1}\sum_{k=0}^{i-j+1}\binom{i-j+1}{k}(-1)^k\widetilde{A}((k+j-1)\mu) \end{aligned}$$

for $i \in \{1, \ldots, N-1\}$, $j \in \{1, \ldots, i+1\}$. The one-step transition probability for $i = N$ and $j \in \mathscr{S}^D$ follows by noting that in that case first one preparation phase has to finish before service can start. Therefore, $P_{N,j} = P_{N-1,j}$ for all $j \in \{1, \ldots, N-1\}$. Finally, transitions corresponding to any other combination of i and j are not possible, leading to a one-step transition probability of zero. Now that the discrete-time Markov chain is constructed, we have that

$$\mathbb{P}(W_n^D > 0) = \mathbb{P}(Z_{n-1}^D = N).$$

Thus, $\mathbb{P}(W_n^D > 0)$, as well as its steady-state version $\lim_{n\to\infty}\mathbb{P}(W_n^D > 0) = \mathbb{P}(W^D > 0)$, can be computed using standard techniques on discrete-time Markov chains. Note that the latter limiting probability indeed exists, since $\{Z_n^D, n \geq 0\}$ is an irreducible and aperiodic Markov chain. Similarly, expressions for the autocorrelation coefficient of consecutive waiting times and the expected number of transitions between two zero waiting times can be computed by analysing this discrete-time Markov chain. This concludes the analysis for exponential preparation times. Conceptually, this analysis can be easily extended to allow for phase-type distribution times at the cost of more cumbersome expressions.

11.3 Ordering of the waiting-time distributions

Now that we know how to compute the waiting-time distribution of the dynamic model for phase-type preparation times, we investigate whether there is any connection between

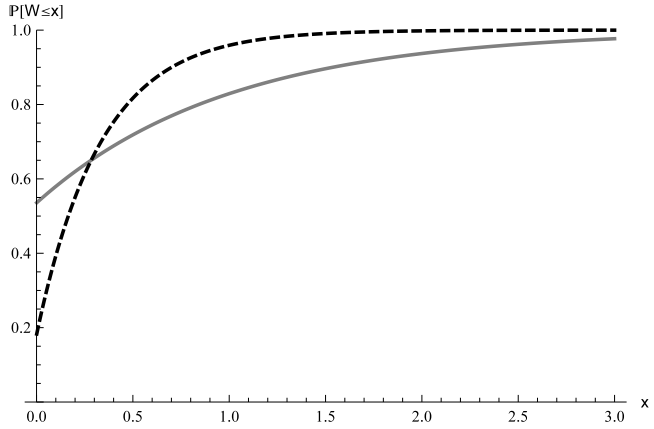

FIGURE 11.1: Waiting-time distribution for the cyclic (solid) and dynamic model (dashed) for $N = 3$ and standard-exponential preparation times. Service times are exponentially (10) distributed.

the waiting-time distributions of both models. In Section 11.3.1, we show that there is not necessarily a stochastic ordering in the two distributions. However, we show in Section 11.3.2 that despite this, the mean waiting time in the dynamic case never exceeds the mean waiting time in the cyclic case.

11.3.1 Stochastic ordering

Intuitively, one might argue that the waiting time W^C of the cyclic system is stochastically larger than or equal to the waiting time W^D of the dynamic system, since one expects that large waiting times occur with higher probability in the cyclic system. In other words, one may conjecture that $\mathbb{P}(W^C > x) \geq \mathbb{P}(W^D > x)$ for all $x \geq 0$. However, this is not necessarily true. One may think of a theoretical setting where the duration of a service time always equals zero. Then, we have for the cyclic case that the n-th waiting time is zero if the preparation time B_n preceding the service of the n-th customer is already completed when the server arrives at the service station. This happens, for example, with positive probability when preparation times are exponentially (μ) distributed (see Section 10.3.1), leading to $\mathbb{P}(W^C > 0) < 1$. In the dynamic case, a zero waiting time could only occur if two preparation phases of different service stations finish at exactly the same point in time. This is, however, not possible, since preparation times are continuously distributed. Hence, we have that $\mathbb{P}(W^D > 0) = 1$, providing a counterexample to the conjecture mentioned above.

This theoretical setting is not the only possible counterexample. Figure 11.1 depicts the waiting-time distributions for both the cyclic and the dynamic case in a system with $N = 3$ service stations, standard-exponential preparation times and exponential (10) service times. This figure shows that a lack of stochastic ordering can occur in a realistic setting, as there clearly exist values of x in this case for which $\mathbb{P}(W^C > x) < \mathbb{P}(W^D > x)$. Of course, there also exist systems for which the waiting times are actually stochastically ordered. For instance, Figure 11.2 shows the waiting-time distributions for the same

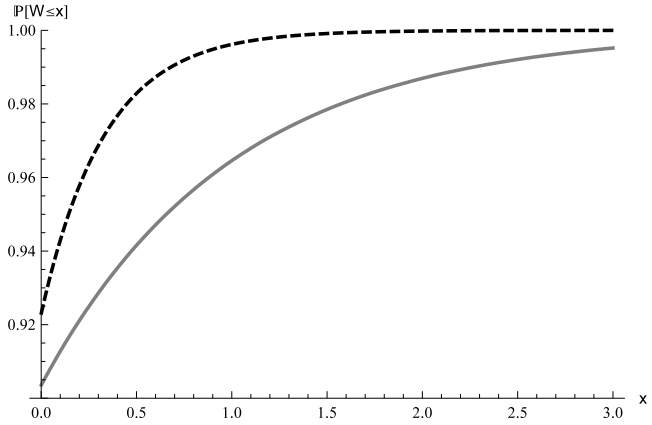

FIGURE 11.2: Waiting-time distribution for the cyclic (solid) and dynamic model (dashed) for $N = 3$ and standard-exponential preparation times. Service times are exponentially (0.5) distributed.

example, except that the service times are now exponentially (0.5) distributed instead. The figure suggests that the waiting-time distributions now never intersect, which implies that they are indeed stochastically ordered. Observe though that a stochastic ordering is not possible in case $N = 2$. It was shown in [266, Theorem 4] that for that case $\mathbb{P}(W^C > 0) \leq \mathbb{P}(W^D > 0)$ for all distributions of A and B and that there does not exist a stochastic ordering for the waiting-time distributions in case preparation times are non-deterministic.

As it is now clear that the waiting-time distributions are not necessarily stochastically ordered, one may still argue that there must at least exist a convex ordering. In other words, one might expect that $\mathbb{E}[\phi(W^C)] \geq \mathbb{E}[\phi(W^D)]$ for any increasing convex function ϕ. If the waiting-time distributions intersect exactly once like in Figure 11.1, the Karlin-Novikoff cut-criterion (cf. [231]) implies that a convex ordering indeed exists. However, the second example in Figure 11.2 shows that there is not always such an intersection, so that the existence of a convex ordering for the general case is hard to prove. Therefore, we focus on the expected waiting times instead in the next section.

11.3.2 Comparison of mean waiting times

Although the waiting-time distributions of the cyclic case and the dynamic case are not necessarily stochastically ordered, one may still reasonably expect that $\mathbb{E}[W^C] \geq \mathbb{E}[W^D]$. In this section, we prove that this weaker conjecture, contrary to the ones in the previous section, holds true for *any* non-negative distribution for A and B by using a sample-path argument. We assume the sequences of realisations $\{b_i, i \geq 1\}$ and $\{a_i, i \geq 1\}$ for the preparation and service times, respectively, to be the same for both scenarios. More specifically, we assume that in both cases the i-th customer that leaves the system does so after having received a service with duration a_i, after which a new customer at the same service station initiates a preparation phase with duration b_i. Furthermore, we assume that when both systems start up, the remaining preparation time of the customer at Q_j at time zero equals ζ_j, $j = 1, \ldots, N$.

To prove that the mean waiting time of the server in the dynamic case does not exceed that of the cyclic case, we require some additional notation. We will denote by $\zeta_{(j)}$ the j-th order statistic of $\zeta_1, \ldots, \zeta_N$, i.e. the j-th smallest value among $\zeta_1, \ldots, \zeta_N$. Let d_i^C be the departure time of the i-th customer after time zero in the cyclic case. The index of the service station at which the server completes a service at time d_i^C in the cyclic case is denoted by q_i^C. Note that $q_i^C = ((i-1) \bmod N) + 1$ for $i > 0$. Furthermore, let $h_{i,j}^C$ be the first moment after d_i^C that a customer at service station $((q_i^C + j - 1) \bmod N) + 1$ has its preparation phase completed and is ready to be served by the server in the cyclic case, $j = 1, \ldots, N-1$.

With these definitions, we obviously have for the first departure that $d_1^C = \zeta_1 + a_1$. Subsequent departures, which are marked by d_i^C, also occur exactly a_i time units after the server starts serving the i-th customer. For $1 < i \leq N-1$ (thus, during the remainder of the first cycle), the start of the i-th service occurs at time $\max\{d_{i-1}^C, \zeta_i\}$, whereas for $i \geq N$ (corresponding to later cycles) the i-th service is initiated at time $\max\{d_{i-1}^C, h_{i-1,1}^C\} = h_{i-1,1}^C$. Therefore,

$$d_i^C = \begin{cases} \zeta_1 + a_1 & \text{if } i = 1, \\ \max\{d_{i-1}^C, \zeta_i\} + a_i & \text{if } 1 < i \leq N-1, \\ h_{i-1,1}^C + a_i & \text{otherwise.} \end{cases} \tag{11.1}$$

As for the h values, we have for $i \leq N-1$ that the first point in time $h_{i,1}^C$ after d_i^C that a customer at Q_{i+1} has its preparation phase completed obviously equals either d_i^C or ζ_{i+1} (whichever happens last). Hence, for $1 \leq i \leq N-1$,

$$h_{i,1}^C = \max\{d_i^C, \zeta_{i+1}\}. \tag{11.2}$$

For $i \geq N$, this expression is more involved. When the server has finished his $(i-1)$-st service, a new preparation phase starts at the corresponding service station while the server moves to the next station. The newly started preparation phase ends at $d_{i-1}^C + b_{i-1}$. It takes $N-1$ additional switches of the server before the customer corresponding to this preparation phase can be served. Hence, $h_{i,N-1}^C$ takes the maximum value of this number and d_i^C. For other values of j, $h_{i,j}^C$ retains the value $h_{i-1,j+1}^C$ corresponding to the situation after the $(i-1)$-st service, in case this value exceeds d_i^C. The shift in the second index is caused because the server has moved one position in the cycle to the next service station between the $(i-1)$-st and the i-th service. To summarise, we thus have for $i \geq N$ that

$$h_{i,j}^C = \begin{cases} \max\{d_i^C, h_{i-1,j+1}^C\} & \text{if } j \neq N-1, \\ \max\{d_i^C, d_{i-1}^C + b_{i-1}\} & \text{if } j = N-1. \end{cases} \tag{11.3}$$

To finalise the notation, let d_i^D, q_i^D and $h_{i,j}^D$ be defined similarly to d_i^C, q_i^C and $h_{i,j}^C$ for the dynamic model. In the dynamic case, the server always moves to the service station with the earliest completed preparation phase. Evidently, we have that $d_1^D = \zeta_{(1)} + a_1$. For $1 < i \leq N-1$, the preparation phase of the i-th served customer finishes before or at time $\zeta_{(i)}$. Therefore, we have for $1 < i \leq N-1$ that

$$d_i^D \leq \max\{d_{i-1}^D, \zeta_{(i)}\} + a_i. \tag{11.4}$$

For values of i larger than $N-1$, we have that the preparation phase the i-th customer goes through has already finished or finishes exactly at time $\min_{j\in\{1,\ldots,N-1\}}\{h^D_{i-1,j}\}$, provided that the $(i-1)$-st customer was served at another station. Otherwise, it obviously finishes at time $d^D_{i-1}+b_i$. Thus, for $i \geq N$, we have

$$d^D_i = \min\{\min_{j\in\{1,\ldots,N-1\}}\{h^D_{i-1,j}\}, d^D_{i-1}+b_i\}+a_i. \tag{11.5}$$

By the definition of $h^D_{i,j}$, it is now not hard to see that for $1 \leq i \leq N-1$,

$$\min_{j\in\{1,\ldots,N-1\}}\{h^D_{i,j}\} \leq \max\{d^D_i, \zeta_{(i+1)}\}. \tag{11.6}$$

For values of i larger than $N-1$, one needs to keep careful track of the position of the server, but otherwise $h^D_{i,j}$ is expressed similarly to (11.3). Namely, for $i \geq N$, we have that

$$h^D_{i,j} = \begin{cases} \max\{d^D_i, h_{i-1,j+((q^D_i-q^D_{i-1}) \bmod N)}\} & \text{if } j \neq N-((q^D_i-q^D_{i-1}) \bmod N), \\ \max\{d^D_i, d^D_{i-1}+b_{i-1}\} & \text{if } j = N-((q^D_i-q^D_{i-1}) \bmod N), \end{cases} \tag{11.7}$$

where $(q^D_i - q^D_{i-1}) \bmod N$ represents the shift in position of the server between time d^D_{i-1} and time d^D_i in the dynamic case.

Now that we have introduced all notation required, we perform two preliminary steps before proving the desired result. First, we show in Lemma 11.3.1 that $d^C_i \geq d^D_i$ for $i = 1,\ldots,N-1$. Thus, we first establish that $d^C_i \geq d^D_i$ for the special case of the first cycle, at the start of which a preparation phase commences at each service point. Then, Lemma 11.3.2 shows that this inequality in fact also holds for $i \geq N$. In other words, the result $d^C_i \geq d^D_i$ persists after the first cycle. Based on these lemmas, Theorem 11.3.3 finally states that $\mathbb{E}[W^C] \geq \mathbb{E}[W^D]$ without any assumption on the distributions of the preparation and service times other than that both distributions have a non-negative support.

LEMMA 11.3.1. *For the first cycle, namely for $i = 1,\ldots,N-1$, we have that*

$$d^C_i \geq d^D_i \text{ and } h^C_{i,1} \geq \min_{j\in\{1,\ldots,N-1\}}\{h^D_{i,j}\}.$$

PROOF. We first focus on the first part of the lemma and prove by induction that $d^C_i \geq d^D_i$ for $i = 1,\ldots,N-1$. We obviously have that

$$d^C_1 = \zeta_1 + a_1 \geq \zeta_{(1)} + a_1 = d^D_1,$$

which acts as a first step of the induction argument. We now show that $d^C_i \geq d^D_i$ for any $1 < i \leq N-1$ under the assumption that $d^C_k \geq d^D_k$ for all $k < i$. More specifically, we conclude based on (11.1) and (11.4) that

$$d^C_i = \max\{d^C_{i-1}, \zeta_i\} + a_i \geq \max\{d^D_{i-1}, \zeta_{(i)}\} + a_i \geq d^D_i$$

for any $1 < i \leq N-1$, by showing that each of the arguments of the second maximum operator does not exceed $\max\{d^C_{i-1}, \zeta_i\}$. To see this for the first argument, note that

$$\max\{d^C_{i-1}, \zeta_i\} \geq d^C_{i-1} \geq d^D_{i-1}$$

by the induction assumption. A similar observation for the second argument follows by noting that

$$\max\{d_{i-1}^C, \zeta_i\} \geq \max\{\max_{j\in\{1,\ldots,i-1\}}\{\zeta_j\}, \zeta_i\} = \max_{j\in\{1,\ldots,i\}}\{\zeta_j\} \geq \zeta_{(i)}.$$

The first inequality holds since d_{i-1}^C must be larger than any of the times $\zeta_1, \ldots, \zeta_{i-1}$, as by time d_{i-1}^C the server has served one customer at the service stations $1, \ldots, i-1$ already in the cyclic case.

For the second part of the lemma, we observe based on (11.2) and (11.6) that for $i = 1, \ldots, N-1$,

$$h_{i,1}^C = \max\{d_i^C, \zeta_{i+1}\} \geq \max\{d_i^D, \zeta_{(i+1)}\} \geq \min_{j\in\{1,\ldots,N-1\}}\{h_{i,j}^D\}. \tag{11.8}$$

The first inequality follows by similar steps to those above. Namely, we obviously have that $\max\{d_i^C, \zeta_{i+1}\} \geq d_i^C \geq d_i^D$ by the first part of the lemma already proved and that

$$\max\{d_i^C, \zeta_{i+1}\} \geq \max\{\max_{j\in\{1,\ldots,i\}}\{\zeta_j\}, \zeta_{i+1}\} = \max_{j\in\{1,\ldots,i+1\}}\{\zeta_j\} \geq \zeta_{(i+1)}.$$

This concludes the proof. □

We now generalise the result obtained in Lemma 11.3.1 and show that $d_i^C \geq d_i^D$ for all $i \geq 1$ in the following lemma.

LEMMA 11.3.2. *At every point in time, namely for every $i \geq 1$, we have that*

$$d_i^C \geq d_i^D \text{ and } h_{i,1}^C \geq \min_{j\in\{1,\ldots,N-1\}}\{h_{i,j}^D\}.$$

PROOF. We have proved this statement already in Lemma 11.3.1 for $i = 1, \ldots, N-1$. To prove the result for larger i, we again apply induction, where Lemma 11.3.1 acts as a first step.

For the induction step, we now prove that $d_i^C \geq d_i^D$ and $h_{i,1}^C \geq \min_{j\in\{1,\ldots,N-1\}}\{h_{i,j}^D\}$ for all $i \geq N$ under the assumption that $d_k^C \geq d_k^D$ and $h_{k,1}^C \geq \min_{j\in\{1,\ldots,N-1\}}\{h_{k,j}^D\}$ for all $k < i$. The former statement $d_i^C \geq d_i^D$ is easily seen to hold true by observing based on (11.1) and (11.5) that

$$d_i^C = h_{i-1,1}^C + a_i \geq \min\{\min_{j\in\{1,\ldots,N-1\}}\{h_{i-1,j}^D\}, d_{i-1}^D + b_i\} + a_i = d_i^D, \tag{11.9}$$

where the inequality holds since $h_{i-1,1}^C \geq \min_{j\in\{1,\ldots,N-1\}}\{h_{i-1,j}^D\}$ as per the induction assumption.

For the latter statement $h_{i,1}^C \geq \min_{j\in\{1,\ldots,N-1\}}\{h_{i,j}^D\}$, we derive from (11.3) that for the cyclic case

$$\begin{aligned} h_{i,1}^C &= \max\{d_i^C, h_{i-1,2}^C\} = \max\{d_i^C, h_{i-2,3}^C\} = \cdots \\ &= \max\{d_i^C, h_{i-N+2,N-1}^C\} = \max\{d_i^C, d_{i-N+1}^C + b_{i-N+1}\}. \end{aligned} \tag{11.10}$$

Similarly, it can be derived from (11.7) that there exist $k, l \in \{1, \ldots, N-1\}$ so that $h_{i,k}^D = \max\{d_i^D, h_{i-N+2,l}^D\}$. This leads to the inequality

$$\min_{j\in\{1,\ldots,N-1\}} h_{i,j}^D \leq \max\{d_i^D, \max_{j\in\{1,\ldots,N-1\}}\{h_{i-N+2,j}^D\}\}. \tag{11.11}$$

We now proceed to show that $h^C_{i,1} \geq \min_{j\in\{1,\ldots,N-1\}}\{h^D_{i,j}\}$ by arguing that $h^C_{i,1}$ is not smaller than each of the arguments in the outer maximum operator in the right-hand side of (11.11). For the first argument, we have by using (11.10) and (11.9), respectively, that

$$h^C_{i,1} = \max\{d^C_i, d^C_{i-N+1} + b_{i-N+1}\} \geq d^C_i \geq d^D_i.$$

To deal with the second argument of the maximum operator, we observe that by (11.7) $\max_{j\in\{1,\ldots,N-1\}}\{h^D_{i-N+2,j}\}$ can evaluate either to a) d^D_{i-N+2}, to b) one of the values from the set $\{d^D_j + b_j : j \in \{1,\ldots,i-N+1\}\}$ or to c) $\zeta_{(N)}$. We treat each of these cases separately below.

a) By (11.10) and (11.9), respectively, we have that

$$h^C_{i,1} = \max\{d^C_i, d^C_{i-N+1} + b_{i-N+1}\} \geq d^C_i \geq d^D_i \geq d^D_{i-N+2}.$$

b) We show that $h^C_{i,1}$ is not smaller than any value in the set $\{d^D_j + b_j : j \in \{1,\ldots,i-N+1\}\}$. To this end, observe that $h^C_{k,1} \geq h^C_{l,1}$ for any $k \geq l$, since

$$h^C_{l,1} \leq d^C_{l+1} \leq d^C_k \leq h^C_{k,1}$$

for all $k > l$. For any $j \in \{1,\ldots,i-N+1\}$, it follows from (11.10), (11.9) and this observation that

$$h^C_{i,1} \geq h^C_{j+N-1,1} = \max\{d^C_{j+N-1}, d^C_j + b_j\} \geq d^C_j + b_j \geq d^D_j + b_j.$$

c) By (11.10) and again the observation that in the cyclic case $h^C_{k,1} \geq h^C_{l,1}$ if $k \geq l$, we have that

$$h^C_{i,1} \geq h^C_{N,1} \geq d^C_{N,1} \geq \zeta_{(N)},$$

where the first inequality again follows from the observation that $h^C_{k,1} \geq h^C_{l,1}$ if $k \geq l$. The second inequality follows from the fact that at time $d^C_{N,1}$, the server has served exactly one customer at each of the service stations, and therefore $d^C_{N,1}$ cannot be smaller than each of the initial residual preparation times $\zeta_1,\ldots,\zeta_N$.

By these observations, we have that $h^C_{i,1} \geq \min_{j\in\{1,\ldots,N-1\}}\{h^D_{i,j}\}$, which concludes the induction step. The lemma now follows by induction on i. □

A combination of Lemmas 11.3.1 and 11.3.2 now leads to the following theorem.

THEOREM 11.3.3. *Given any two non-negative distributions for the service time A and the preparation time B, we have that* $\mathbb{E}[W^C] \geq \mathbb{E}[W^D]$.

PROOF. Given any two sets of independent and identically distributed sequences $\{a_i, i \geq 1\}$ and $\{b_i, i \geq 1\}$ from the random variables A and B, and any initial set of preparation times $(\zeta_1,\ldots,\zeta_N)$, Lemma 11.3.2 states that $d^C_i \geq d^D_i$ for all $i \geq 1$.

Observe that $d^C_i = \sum_{j=1}^{i}(w^C_j + a_j)$, where w^C_j is the time the server has to wait directly before the start of the j-th service in the cyclic scenario. Likewise, we have that $d^D_i =$

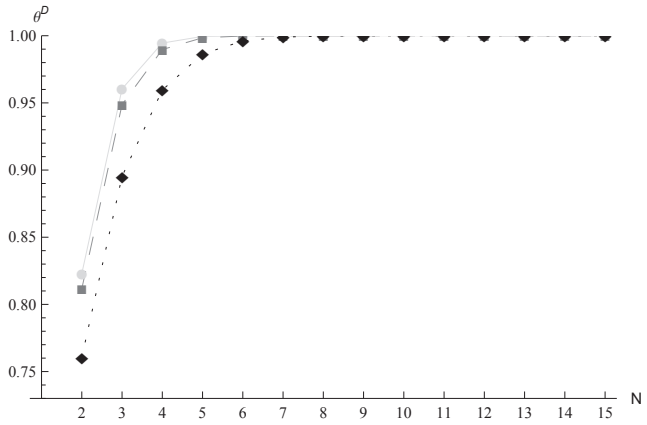

FIGURE 11.3: Throughput as a function of the number of stations for moderately variable preparation and service times (solid), highly variable service times (dotted) and highly variable preparation times (dashed) in the dynamic model.

$\sum_{j=1}^{i}(w_j^D + a_j)$, where w_i^D is defined similarly to w_i^C for the dynamic scenario. Therefore, the lemma implies that for all $i > 0$,

$$\sum_{j=1}^{i}(w_j^C + a_j) \geq \sum_{j=1}^{i}(w_j^D + a_j), \tag{11.12}$$

which, after subtracting $\sum_{j=1}^{i} a_j$, dividing by i and taking limits on both sides, leads to

$$\lim_{i\to\infty} \frac{\sum_{j=1}^{i} w_j^C}{i} \geq \lim_{i\to\infty} \frac{\sum_{j=1}^{i} w_j^D}{i}.$$

The left-hand side (right-hand side) represents the asymptotic mean waiting time of the server in the cyclic (dynamic) scenario given the realisations $\{b_i, i \geq 1\}$, $\{a_i, i \geq 1\}$ and $(\zeta_1, \ldots, \zeta_N)$. Therefore, the theorem follows by conditioning on these realisations. □

REMARK 11.3.1. It is suggested by (11.12) that $\sum_{j=1}^{i} W_j^C$ is stochastically larger than or equal to $\sum_{j=1}^{i} W_j^D$ for all $i > 0$, where W_j^C (W_j^D) is the random variable representing the j-th waiting time of the server in the cyclic (dynamic) case. Although there is not necessarily a stochastic ordering in the limiting distributions of the waiting times W^C and W^D (cf. Section 11.3.1), it thus appears that there exists a stochastic ordering in partial sums of transient waiting times starting at $j = 1$.

11.4 Numerical comparison

In Section 10.4, we gained several insights into the effect of the system parameters on its performance in the cyclic model. More specifically, we commented on the effect of variability of the preparation and service times, we studied the autocorrelation coefficients of

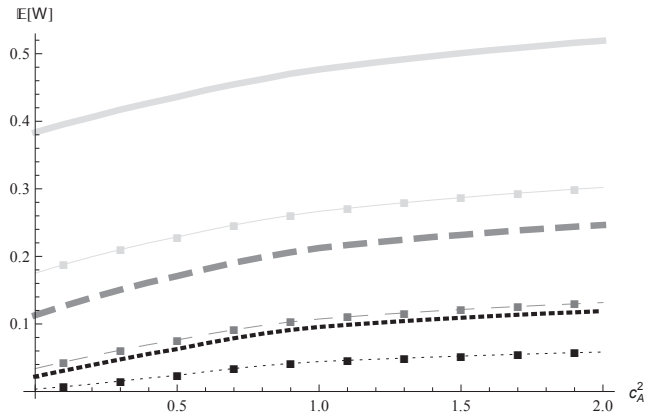

FIGURE 11.4: Mean waiting time as a function of c_A^2 for the cyclic (thick) and dynamic (thin and marked) model with the values $r = 0.5$ (solid), $r = 0.8$ (dashed) and $r = 1.2$ (dotted).

the waiting times and we studied the number of stations to be assigned to a server. In this section, we compare the insights gained for the cyclic model with equivalent observations for the dynamic model based on additional simulation results, and we explicitly comment on similarities and differences between the two models.

Variability of preparation and service times We observed in Section 10.4 that the variability of the preparation time in the cyclic model seems to have a bigger impact on the server's waiting-time process than the variability of the service times. This observation does not extend to the dynamic case. Although the impact of the variability of the service times is similar, the variability of the preparation times hardly seems to matter for the waiting-time process. In Figure 11.3, we have plotted the counterpart of Figure 10.1 where the server now visits the service stations dynamically rather than cyclically. Thus, for the same variability settings considered before, we now plot the throughput θ^D versus the number of queues N.

It turns out that the solid curve and the dotted curve corresponding to moderately variable preparation times are similar to the ones corresponding to the cyclic model, except that, as expected, these curves converge faster to the maximum throughput. However, whereas the dashed curve corresponding to highly variable preparation times was farthest away from the solid curve in Figure 10.1, the solid and dashed curves now almost coincide. This indicates that the variability of the preparation times hardly matters for the server's waiting time in the dynamic model. This can be explained by the fact that the dynamic model has many similarities with the Erlang loss model. In fact, if the service time A were exponentially distributed, the dynamic model would reduce to an M/G/N/N queueing system. The service completions in the dynamic model are then equivalent to Poisson arrivals in the M/G/N/N queue, of which the number of customers present represents the number of preparations in progress. A distinctive feature of the M/G/N/N queue is that its performance measures are insensitive to the distribution of B apart from its first moment (see e.g. [138]). Thus, if we would have chosen exponential service times, the solid curve and the dashed curve would have coincided. As this is not the case

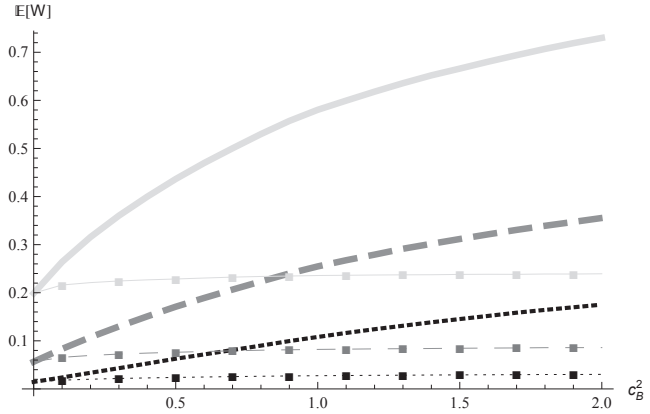

FIGURE 11.5: Mean waiting time as a function of c_B^2 for the cyclic (thick) and dynamic (thin and marked) model with the values $r = 0.5$ (solid), $r = 0.8$ (dashed) and $r = 1.2$ (dotted).

in our current example, the curves do not completely coincide, but the throughput of the system nevertheless seems hardly sensitive to the distribution of B.

To further study the effects of the variability of the two time components, we define the squared coefficient of variation $c_A^2 = \text{Var}[A]/(\mathbb{E}[A])^2$. Let c_B^2 be defined similarly, and let $r = \mathbb{E}[B]/\mathbb{E}[A]$ represent the ratio of the two time components. Consider the systems with $N = 3$, $\mathbb{E}[A] = 1$ and the values $r = 0.5$, $r = 0.8$ and $r = 1.2$. Figures 11.4 and 11.5 plot the mean waiting time $\mathbb{E}[W]$ versus c_A^2 (keeping c_B^2 fixed at 1.5) and c_B^2 (keeping c_A^2 fixed at 1.5), respectively. In these two graphs, thick lines correspond to the cyclic case, whereas the thin, marked lines indicate results where the server visits the stations dynamically. From Figure 11.4, we conclude that as c_A^2 increases, the mean waiting time also increases for both cases, but that the rate of change is bigger in the cyclic case. However, the difference between a curve corresponding to the dynamic case and its equivalent for the cyclic case is eventually almost constant, and this difference increases as the value of r decreases. In Figure 11.5, we see that the mean waiting time in the cyclic model is more sensitive to c_B^2 than c_A^2 as observed before. However, for the dynamic system it is indeed almost insensitive to c_B^2. Finally, we observe that in case $c_B^2 = 0$ (i.e. deterministic preparation times), the mean waiting times for the cyclic and the dynamic model coincide. Since deterministic preparation phases will always complete in the order they were initiated, the server will also serve the service points in a fixed cyclic order in the dynamic case, which leads to this behaviour.

Correlations In Section 10.4, we observed that the stationary autocorrelation coefficients of the waiting times in the cyclic model show a rather surprising behaviour. The stationary autocorrelation coefficients pertaining to the dynamic model turn out to behave just as surprisingly, but they show a behaviour completely different from the cyclic case. In Figures 11.6, 11.7 and 11.8, we plot the stationary autocorrelation coefficients of the waiting times in the dynamic model based on the same system settings as those used to construct Figure 10.3, namely exponentially (1) distributed preparation times, exponentially (10) distributed service times and $N = 5$. However, apart from the exponential

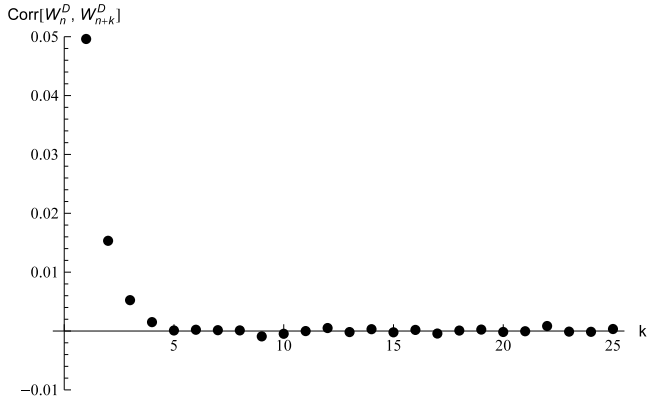

FIGURE 11.6: Stationary autocorrelation coefficients of the waiting times in the dynamic model with $c_B^2 = 1$.

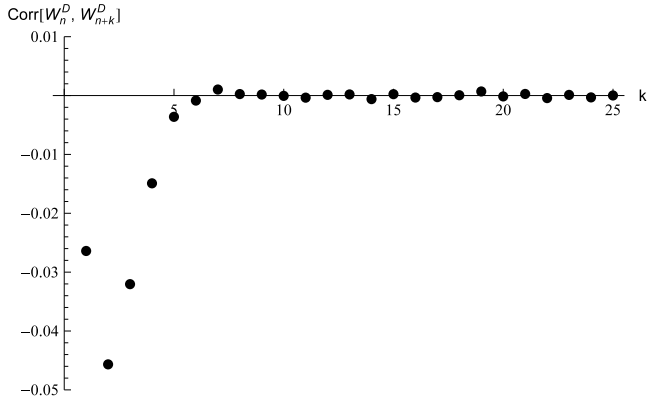

FIGURE 11.7: Stationary autocorrelation coefficients of the waiting times in the dynamic model with $c_B^2 = 0.5$.

case $c_B^2 = 1$ in Figure 11.6, we now also regard the autocorrelation coefficients for the values $c_B^2 = 0.5$ and $c_B^2 = 10$ in Figures 11.7 and 11.8, respectively.

In the cyclic case, increasing the value of c_B^2 does not alter the shape of the curve depicted in Figure 10.3, although the correlation generally becomes less significant. Figures 11.6, 11.7 and 11.8 show not only that the correlation becomes more significant and converges to zero slower in the dynamic case as c_B^2 increases, but also that the shape of the curve is sensitive to c_B^2. Figures 11.6 and 11.7 clearly show that also in the dynamic model periodicity effects are present, as alternatingly convex and concave loops can be observed. However, an increasing c_B^2 also seems to have a significant effect on the correlation itself. For $c_B^2 = 0.5$, the correlation is negative for small k, whereas this is not the case for $c_B^2 = 1.0$. For $c_B^2 = 10$, Figure 11.8 even shows a monotonously decreasing curve. It is not clear why these effects are present. The significant influence of the variability of the preparation times on the autocorrelation is highly surprising, as we observed that the waiting-time distribution itself is hardly sensitive to c_B^2. Such peculiar behaviour is also

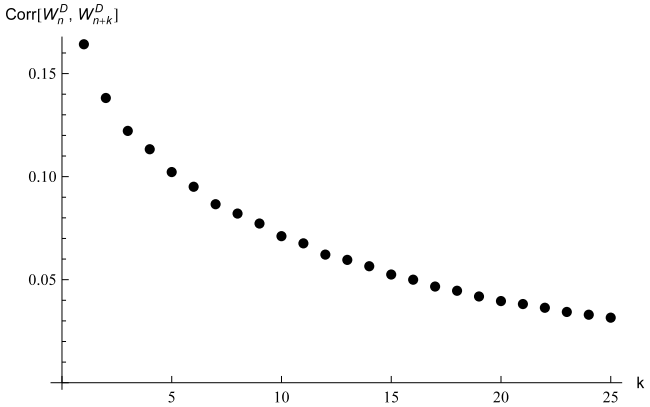

FIGURE 11.8: Stationary autocorrelation coefficients of the waiting times in the dynamic model with $c_B^2 = 10$.

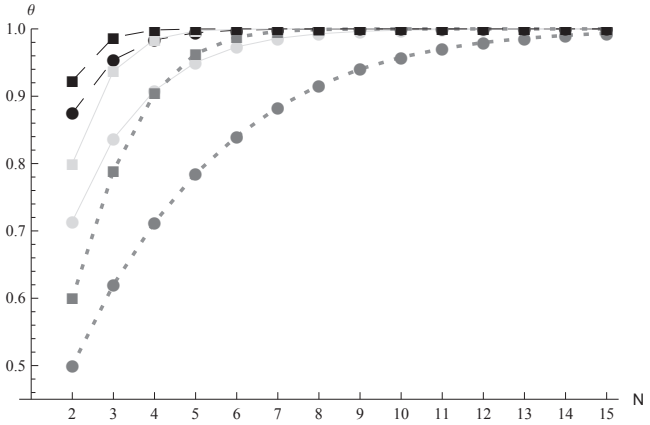

FIGURE 11.9: Throughput as a function of the number of stations for small (dashed), moderate (solid) and large preparation times (dotted) for the cyclic (circles) and the dynamic model (squares).

present for the variability of the service time, but in an opposite fashion. Whereas the waiting-time distribution is sensitive to c_A^2 in the dynamic case (cf. Figure 11.4), numerical results show that this number has little effect on the correlation curves as depicted in Figures 11.6, 11.7 and 11.8.

Number of stations to be assigned to a server We now study how the number of stations to be assigned to a server changes when one switches from a cyclic to a dynamic regime. In Figure 11.9, we plot the same curves as those depicted in Figure 10.4, and we add the curves one would obtain when the server visits the service stations dynamically. This figure shows intuitive results. Obviously, the throughput θ^D for the dynamic model is larger than its equivalent θ^C for the cyclic model. This is not surprising, since we found in Section 11.3.2 that $\mathbb{E}[W^C] \geq \mathbb{E}[W^D]$. As a result, the number of stations to be assigned

to a server in order to be close to maximum throughput decreases. Whereas we concluded before that generally about 5 or 6 servers are needed for the cyclic case, it seems that for the dynamic case about 3 to 4 servers are already enough.

BIBLIOGRAPHY

Self-references

[P1] BEKKER, R., DORSMAN, J. L., VAN DER MEI, R. D., VIS, P. AND WINANDS, E. M. M. (2013). Scheduling in polling systems in heavy traffic. *ACM SIGMETRICS Performance Evaluation Review* **41,** 41–43. Special issue on the 31st International Symposium on Computer Performance, Modeling, Measurements and Evaluation (IFIP WG 7.3 Performance 2013).

[P2] BEKKER, R., VIS, P., DORSMAN, J. L., VAN DER MEI, R. D. AND WINANDS, E. M. M. (2015). The impact of scheduling policies on the waiting-time distributions in polling systems. *Queueing Systems*. To appear.

[P3] DORSMAN, J. L., BHULAI, S. AND VLASIOU, M. (2015). Dynamic server assignment in an extended machine-repair model. *IIE Transactions*. To appear.

[P4] DORSMAN, J. L., BORST, S. C., BOXMA, O. J. AND VLASIOU, M. (2014). Markovian polling systems with an application to wireless random-access networks. *Technical report* 2014-001. Eurandom Preprint Series. Submitted.

[P5] DORSMAN, J. L., BOXMA, O. J. AND VAN DER MEI, R. D. (2014). On two-queue Markovian polling systems with exhaustive service. *Queueing Systems* **78,** 287–311.

[P6] DORSMAN, J. L., BOXMA, O. J. AND VLASIOU, M. (2013). Marginal queue length approximations for a two-layered network with correlated queues. *Queueing Systems* **75,** 29–63.

[P7] DORSMAN, J. L., PEREL, N. AND VLASIOU, M. (2013). Server waiting times in infinite supply polling systems with preparation times. *Technical report* 2013-018. Eurandom Preprint Series. Submitted.

[P8] DORSMAN, J. L., VAN DER MEI, R. D. AND VLASIOU, M. (2013). Analysis of a two-layered network by means of the power-series algorithm. *Performance Evaluation* **70,** 1072–1089.

[P9] DORSMAN, J. L., VAN DER MEI, R. D. AND WINANDS, E. M. M. (2011). A new method for deriving waiting-time approximations in polling systems with renewal arrivals. *Stochastic Models* **27,** 318–332.

[P10] DORSMAN, J. L., VAN DER MEI, R. D. AND WINANDS, E. M. M. (2012). Polling systems with batch service. *OR Spectrum* **34,** 743–761.

[P11] DORSMAN, J. L., VLASIOU, M. AND ZWART, B. (2013). Parallel queueing networks with Markov-modulated service speeds in heavy traffic. *ACM SIGMETRICS Performance Evaluation Review* **41,** 47–49. Special issue on the 31st International Symposium on Computer Performance, Modeling, Measurements and Evaluation (IFIP WG 7.3 Performance 2013).

[P12] DORSMAN, J. L., VLASIOU, M. AND ZWART, B. (2015). Heavy-traffic asymptotics for networks of parallel queues with Markov-modulated service speeds. *Queueing Systems*. To appear.

[P13] PEREL, N., DORSMAN, J. L. AND VLASIOU, M. (2013). Cyclic-type polling models with preparation times. In *Proceedings of the 2nd International Conference on Operations Research and Enterprise Systems*. pp. 14–23. **Received the Best Paper Award**.

[P14] ROGIEST, W., DORSMAN, J. L. AND FIEMS, D. (2014). Analysis of fibre-loop optical buffers with a void-avoiding schedule. In *Proceedings of the 8th International Conference on Performance Evaluation Methodologies and Tools (Valuetools 2014)*.

References

[15] AL-OMARI, T., DERISAVI, S. AND FRANKS, R. G. (2007). Deriving distribution of thread service time in layered queueing networks. In *Proceedings of the 6th International Workshop on Software and Performance*. pp. 66–77.

[16] AL-OMARI, T., FRANKS, R. G., WOODSIDE, C. M. AND PAN, A. (2005). Solving layered queueing networks of large client-server systems with symmetric replication. In *Proceedings of the 5th International Workshop on Software and Performance*. pp. 159–166.

[17] AL-OMARI, T., FRANKS, R. G., WOODSIDE, C. M. AND PAN, A. (2007). Efficient performance models for layered server systems with replicated servers and parallel behaviour. *Journal of Systems and Software* **80,** 510–527.

[18] ALTMAN, E. (2002). Stochastic recursive equations with applications to queues with dependent vacations. *Annals of Operations Research* **112,** 43–61.

[19] ASMUSSEN, S. (2003). *Applied Probability and Queues*. Springer, New York.

[20] ATA, B. AND SHNEORSON, S. (2006). Dynamic control of an M/M/1 service system with adjustable arrival and service rates. *Management Science* **52,** 1778–1791.

[21] ATHREYA, K. B. AND NEY, P. E. (1972). *Branching Processes*. Springer, Berlin/Heidelberg.

[22] AVI-ITZHAK, B. AND HALFIN, S. (1988). Expected response times in a non-symmetric time sharing queue with a limited number of service positions. In *Proceedings of the 12th International Teletraffic Congress*. pp. 1485–1493.

[23] BACCELLI, F. AND BRÉMAUD, P. (2003). *Elements of Queueing Theory: Palm-Martingale Calculus and Stochastic Recurrences*. Springer, New York.

[24] BACIGALUPO, D. A., JARVIS, S. A., HE, L., SPOONER, D. P., DILLENBERGER, D. N. AND NUDD, G. R. (2005). An investigation into the application of different performance prediction methods to distributed enterprise applications. *The Journal of Supercomputing* **34,** 93–111.

[25] BACIGALUPO, D. A., JARVIS, S. A., HE, L., SPOONER, D. P. AND NUDD, G. R. (2005). Comparing layered queuing and historical performance models of a distributed enterprise application. In *Proceedings of the IASTED International Conference on Parallel and Distributed Computing and Networks*. pp. 608–613.

[26] BACIGALUPO, D. A., VAN HEMERT, J., CHEN, X., USMANI, A., CHESTER, A. P., HE, L., DILLENBERGER, D. N., WILLS, G. B., GILBERT, L. AND JARVIS, S. A. (2011). Managing dynamic enterprise and urgent workloads on clouds using layered queuing and historical performance models. *Simulation Modelling Practice and Theory* **19,** 1479–1495.

[27] BACIGALUPO, D. A., VAN HEMERT, J., USMANI, A., DILLENBERGER, D. N., WILLS, G. B. AND JARVIS, S. A. (2010). Resource management of enterprise cloud systems using layered queuing and historical performance models. In *Proceedings of the 24th IEEE International Symposium on Parallel and Distributed Processing*. pp. 1–8.

[28] BALSAMO, S. (2011). Queueing networks with blocking: Analysis, solution algorithms and properties. In *Network Performance Engineering*. Ed. D. D. Kouvatsos. Vol. 5233 of *Lecture Notes in Computer Science*. Springer, Berlin/Heidelberg. pp. 233–257.

[29] BARD, Y. (1979). Some extensions to multiclass queueing network analysis. In *Proceedings of the 3rd International Symposium on Modelling and Performance Evaluation of Computer Systems*. pp. 51–62.

[30] BASKETT, F., CHANDY, K. M., MUNTZ, R. R. AND PALACIOS, F. G. (1975). Open, closed, and mixed networks of queues with different classes of customers. *Journal of the ACM* **22,** 248–260.

[31] BEKKER, R. (2005). Queues with State-Dependent Rates. *PhD thesis*. Eindhoven University of Technology, Eindhoven, The Netherlands.

[32] BERTOIN, J. (1996). *Lévy Processes*. Cambridge University Press, Cambridge.

[33] BERTSEKAS, D. AND GALLAGER, R. (1992). *Data Networks*. Prentice Hall, Englewood Cliffs.

[34] BERTSIMAS, D. AND MOURTZINOU, G. (1999). Decomposition results for general polling systems and their applications. *Queueing Systems* **31,** 295–316.

[35] BHULAI, S. (2006). On the value function of the M/Cox(r)/1 queue. *Journal of Applied Probability* **43,** 363–376.

[36] BHULAI, S. (2009). Dynamic routing policies for multiskill call centers. *Probability in the Engineering and Informational Sciences* **23,** 101–119.

[37] BHULAI, S. AND SPIEKSMA, F. M. (2003). On the uniqueness of solutions to the Poisson equations for average cost Markov chains with unbounded cost functions. *Mathematical Methods of Operations Research* **58,** 221–236.

[38] BINGHAM, N. H., GOLDIE, C. M. AND TEUGELS, J. L. (1989). *Regular Variation*. Cambridge University Press, Cambridge.

[39] BISH, E. K., LEONG, T., LI, C., NG, J. W. C. AND SIMCHI-LEVI, D. (2001). Analysis of a new vehicle scheduling and location problem. *Naval Research Logistics* **48,** 363–385.

[40] BLANC, J. P. C. (1987). A note on waiting times in systems with queues in parallel. *Journal of Applied Probability* **24,** 540–546.

[41] BLANC, J. P. C. (1987). On a numerical method for calculating state probabilities for queueuing systems with more than one waiting line. *Journal of Computation and Applied Mathematics* **20,** 119–125.

[42] BLANC, J. P. C. (1993). Performance analysis and optimization with the power-series algorithm. In *Performance Evaluation of Computer and Communication Systems*. Eds. L. Donatiello and R. D. Nelson. Vol. 729 of *Lecture Notes in Computer Science*. Springer, Berlin/Heidelberg. pp. 53–80.

[43] BOON, M. A. A., VAN DER MEI, R. D. AND WINANDS, E. M. M. (2011). Applications of polling systems. *Surveys in Operations Research and Management Science* **16,** 67–82.

[44] BOON, M. A. A. AND WINANDS, E. M. M. (2014). Heavy-traffic analysis of k-limited polling systems. *Probability in the Engineering and Informational Sciences* **28,** 451–471.

[45] BOON, M. A. A., WINANDS, E. M. M., ADAN, I. J. B. F. AND VAN WIJK, A. C. C. (2011). Closed-form waiting time approximations for polling systems. *Performance Evaluation* **68,** 290–306.

[46] BOXMA, O. J. (1989). Workloads and waiting times in single-server systems with multiple customer classes. *Queueing Systems* **5,** 185–214.

[47] BOXMA, O. J. AND COMBÉ, M. B. (1993). The correlated M/G/1 queue. *Archiv für Elektronik und Übertragungstechnik* **47,** 330–335.

[48] BOXMA, O. J. AND DADUNA, H. (1990). Sojourn times in queueing networks. In *Stochastic Analysis of Computer and Communication Systems*. Ed. H. Takagi. North-Holland, Amsterdam. pp. 401–450.

[49] BOXMA, O. J. AND GROENENDIJK, W. P. (1987). Pseudo-conservation laws in cyclic-service systems. *Journal of Applied Probability* **24,** 949–964.

[50] BOXMA, O. J. AND GROENENDIJK, W. P. (1988). Two queues with alternating service and switching times. In *Queueing Theory and its Applications (Liber Amicorum for J. W. Cohen)*. Eds. O. J. Boxma and R. Syski. North-Holland, Amsterdam. pp. 261–282.

[51] BOXMA, O. J., KELLA, O. AND KOSIŃSKI, K. M. (2011). Queue lengths and workloads in polling systems. *Operations Research Letters* **39,** 401–405.

[52] BOXMA, O. J., LEVY, H. AND WEISTSTRATE, J. A. (1990). Optimization of polling systems. In *Proceedings of the 14th IFIP WG 7.3 International Symposium on Computer Performance Modelling, Measurement and Evaluation*. pp. 349–361.

[53] BOXMA, O. J., MANDJES, M. R. H. AND KELLA, O. (2008). On a queueing model with service interruptions. *Probability in the Engineering and Informational Sciences* **22,** 537–555.

[54] BOXMA, O. J. AND WEISTSTRATE, J. A. (1989). Waiting times in polling systems with Markovian server routing. In *Messung, Modellierung und Bewertung von Rechensystemen und Netzen*. Eds. G. Stiege and J. S. Lie. Springer, Berlin/Heidelberg. pp. 89–104.

[55] BOYD, S. AND VANDENBERGHE, L. (2004). *Convex Optimization*. Cambridge University Press, Cambridge.

[56] BUDHIRAJA, A. AND LEE, C. (2009). Stationary distribution convergence for generalized Jackson networks in heavy traffic. *Mathematics of Operations Research* **34,** 45–56.

[57] BUHR, R. J. A. (1998). Use case maps as architectural entities for complex systems. *IEEE Transactions on Software Engineering* **24,** 1131–1155.

[58] CHANDY, K. M. AND NEUSE, D. (1982). Linearizer: A heuristic algorithm for queueing network models of computing systems. *Communications of the ACM* **25,** 126–134.

[59] CHEN, H. AND WHITT, W. (1993). Diffusion approximations for open queueing networks with service interruptions. *Queueing Systems* **13,** 335–359.

[60] CHEN, H. AND YAO, D. D. (2001). *Fundamentals of Queueing Networks*. Springer, New York.

[61] CHIANG, M., LOW, S. H., CALDERBANK, A. R. AND DOYLE, J. C. (2007). Layering as optimization decomposition: A mathematical theory of network architectures. *Proceedings of the IEEE* **95,** 255–312.

[62] CHOUDHURY, G. L., MANDELBAUM, A., REIMAN, M. I. AND WHITT, W. (1997). Fluid and diffusion limits for queues in slowly changing environments. *Communications in Statistics. Stochastic Models* **13,** 121–146.

[63] CHUNG, H., UN, C. K. AND JUNG, W. Y. (1994). Performance analysis of Markovian polling systems with single buffers. *Performance Evaluation* **19,** 303–315.

[64] CLINE, D. B. H. AND SAMORODNITSKY, G. (1994). Subexponentiality of the product of independent random variables. *Stochastic Processes and their Applications* **49,** 75–98.

[65] COFFMAN JR., E. G., PUHALSKII, A. A. AND REIMAN, M. I. (1995). Polling systems with zero switch-over times: A heavy-traffic principle. *Annals of Applied Probability* **5,** 681–719.

[66] COFFMAN JR., E. G., PUHALSKII, A. A. AND REIMAN, M. I. (1998). Polling systems in heavy-traffic: A Bessel process limit. *Mathematics of Operations Research* **23,** 257–304.

[67] COHEN, J. W. (1982). *The Single Server Queue*. North-Holland, Amsterdam.

[68] COOPER, R. B. (1970). Queues served in cyclic order: Waiting times. *Bell System Technical Journal* **49,** 399–413.

[69] COOPER, R. B. AND MURRAY, G. (1969). Queues served in cyclic order. *Bell System Technical Journal* **48,** 675–689.

[70] DAI, J. G. AND HARRISON, J. M. (1992). Reflected Brownian motion in an orthant: Numerical methods for steady-state analysis. *The Annals of Applied Probability* **2,** 65–86.

[71] D'AMBROGIO, A. (2005). A model transformation framework for the automated building of performance models from UML models. In *Proceedings of the 5th International Workshop on Software and Performance*. pp. 75–86.

[72] D'AMBROGIO, A. AND BOCCIARELLI, P. (2007). A model-driven approach to describe and predict the performance of composite services. In *Proceedings of the 6th International Workshop on Software and Performance*. pp. 78–89.

[73] DAS, O. AND WOODSIDE, C. M. (2001). Failure detection and recovery modelling for multi-layered service systems. In *Proceedings of the 5th International Workshop on Performability Modeling of Computer and Communication Systems*. pp. 131–135.

[74] DAS, O. AND WOODSIDE, C. M. (2002). Modeling the coverage and effectiveness of fault-management architectures in layered distributed systems. In *Proceedings of the International Conference on Dependable Systems and Networks*. pp. 745–754.

[75] DAS, O. AND WOODSIDE, C. M. (2003). The influence of layered system structure on strategies for software rejuvenation. In *Proceedings of the 6th International Workshop on Performability Modeling of Computer and Communication Systems*. pp. 47–50.

[76] DAS, O. AND WOODSIDE, C. M. (2004). Computing the performability of layered distributed systems with a management architecture. In *Proceedings of the 4th International Workshop on Software and Performance*. pp. 174–185.

[77] DAS, O. AND WOODSIDE, C. M. (2004). Dependability modeling of self-healing client-server applications. In *Architecting Dependable Systems II*. Eds. R. de Lemos, C. Gacek, and A. Romanovsky. Vol. 3069 of *Lecture Notes in Computer Science*. Springer, Berlin/Heidelberg. pp. 266–285.

[78] DEBICKI, K., KOSIŃSKI, K. M. AND MANDJES, M. R. H. (2012). Gaussian queues in light and heavy traffic. *Queueing Systems* **71,** 137–149.

[79] DEKKER, R., VOOGD, P. AND VAN ASPEREN, E. (2007). Advanced methods for container stacking. In *Container Terminals and Cargo Systems*. Eds. K. H. Kim and H. Günther. Springer, Berlin/Heidelberg. pp. 131–154.

[80] DELCOIGNE, F. AND DE LA FORTELLE, A. (2002). Large deviations rate function for polling systems. *Queueing Systems* **41,** 13–44.

[81] DI MARCO, A. AND MASCOLO, C. (2007). Performance analysis and prediction of physically mobile systems. In *Proceedings of the 6th International Workshop on Software and Performance*. pp. 129–132.

[82] DIEKER, A. B. AND MORIARTY, J. (2009). Reflected Brownian motion in a wedge: Sum-of-exponential stationary densities. *Electronic Communications in Probability* **14,** 1–16.

[83] DILLEY, J., FRIEDRICH, R., JIN, T. AND ROLIA, J. A. (1998). Web server performance measurement and modeling techniques. *Performance Evaluation* **33,** 5–26.

[84] DOSHI, B. T. (1986). Queueing systems with vacations: A survey. *Queueing Systems* **1,** 29–66.

[85] DOSHI, B. T. (1990). Single server queues with vacations. In *Stochastic Analysis of Computer and Communication Systems*. Ed. H. Takagi. North-Holland, Amsterdam. pp. 217–265.

[86] DUINKERKEN, M. B., DEKKER, R., KURSTJENS, S. T. G. L., OTTJES, J. A. AND DELLAERT, N. P. (2007). Comparing transportation systems for inter-terminal transport at the Maasvlakte container terminals. In *Container Terminals and Cargo Systems*. Eds. K. H. Kim and H. Günther. Springer, Berlin/Heidelberg. pp. 37–61.

[87] EISENBERG, M. (1972). Queues with periodic service and changeover time. *Operations Research* **20,** 440–451.

[88] EISENBERG, M. (1979). Two queues with alternating service. *SIAM Journal on Applied Mathematics* **36,** 287–303.

[89] EL-DESOUKY, A. I., ALI, H. A. AND ABDUL-AZEEM, Y. M. (2008). LQN-based performance evaluation framework of UML-based models for distributed object applications.

[90] EL-KAEDY, R. A. AND SAMEH, A. (2011). Integrating performance characterization with software development. *International Journal of Basic & Applied Sciences* **11,** 7–13.

[91] EL-KAEDY, R. A. AND SAMEH, A. (2011). Performance analysis and characterization tool for distributed software development. *International Journal of Research and Reviews in Computer Science* **2,** 906–915.

[92] EL-SAYED, H., CAMERON, D. AND WOODSIDE, C. M. (1998). Automated performance modeling from scenarios and SDL designs of distributed systems. In *Proceedings of the International Symposium on Software Engineering for Parallel and Distributed Systems*. pp. 127–135.

[93] ELIAZAR, I. (2005). Gated polling systems with Lévy inflow and inter-dependent switchover times: A dynamical-systems approach. *Queueing Systems* **49,** 49–72.

[94] ERLANG, A. K. (1909). The theory of probabilities and telephone conversations. *Nyt Tidsskrift for Matematik* **20,** 33–39.

[95] FAYOLLE, G. AND LASGOUTTES, J. M. (1995). A state-dependent polling model with Markovian routing. In *The IMA volumes in Mathematics and Its Applications*. Eds. F. P. Kelly and R. J. Williams. Springer, Berlin/Heidelberg. pp. 283–311.

[96] FELLER, W. (1971). *An Introduction to Probability Theory and its Applications, Vol. II*. Wiley, New York.

[97] FENG, W., OHI, F. AND KOWADA, M. (2006). Large deviations of Markovian polling models with applications to admission control. *Annals of Operations Research* **146,** 169–188.

[98] FIEMS, D. AND ALTMAN, E. (2012). Gated polling with stationary ergodic walking times, Markovian routing and random feedback. *Annals of Operations Research* **198,** 145–164.

[99] FLEMING, P. J. AND SIMON, B. (1991). Interpolation approximations of sojourn time distributions. *Operations Research* **39,** 251–260.

[100] FRANKS, R. G., AL-OMARI, T., WOODSIDE, C. M., DAS, O. AND DERISAVI, S. (2009). Enhanced modeling and solution of layered queueing networks. *IEEE Transactions on Software Engineering* **35,** 148–161.

[101] FRANKS, R. G. AND WOODSIDE, C. M. (2004). Multiclass multiservers with deferred operations in layered queueing networks, with software system applications. In *Proceedings of the 12th IEEE/ACM International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems*. pp. 239–248.

[102] FUHRMANN, S. W. AND COOPER, R. B. (1985). Stochastic decompositions in the M/G/1 queue with generalized vacations. *Operations Research* **33,** 1117–1129.

[103] GAMARNIK, D., NOWICKI, T. AND SWIRSZCZ, G. (2006). Maximum weight independent sets and matchings in sparse random graphs. Exact results using the local weak convergence method. *Random Structures & Algorithms* **28,** 76–106.

[104] GAMARNIK, D. AND ZEEVI, A. (2006). Validity of heavy traffic steady-state approximation in generalized Jackson networks. *The Annals of Applied Probability* **16,** 56–90.

[105] GEORGE, J. M. AND HARRISON, J. M. (2001). Dynamic control of a queue with adjustable service rate. *Operations Research* **49,** 720–731.

[106] GORDON, W. AND NEWELL, G. (1967). Closed queuing systems with exponential servers. *Operations Research* **15,** 254–265.

[107] GRADL, S., BÖGELSACK, A., WITTGES, H. AND KRCMAR, H. (2009). Layered queuing networks for simulating enterprise resource planning systems. In *Proceedings of the 7th International Workshop on Modelling, Simulation, Verification and Validation of Enterprise Information Systems*. pp. 85–92.

[108] GROENEVELT, R. AND ALTMAN, E. (2005). Analysis of alternating-priority queueing models with (cross) correlated switchover times. *Queueing Systems* **51,** 199–247.

[109] GROSS, D. AND INCE, J. F. (1981). The machine repair problem with heterogeneous populations. *Operations Research* **29,** 532–549.

[110] GROSSGLAUSER, M. AND TSE, D. (2002). Mobility increases the capacity of ad hoc wireless networks. *IEEE/ACM Transactions on Networking* **10,** 477–486.

[111] GU, G. P. AND PETRIU, D. C. (2002). XSLT transformation from UML models to LQN performance models. In *Proceedings of the 3rd International Workshop on Software and Performance*. pp. 227–234.

[112] GU, G. P. AND PETRIU, D. C. (2005). From UML to LQN by XML algebra-based model transformations. In *Proceedings of the 5th International Workshop on Software and Performance*. pp. 99–110.

[113] HAIJEMA, R. AND VAN DER WAL, J. (2008). An MDP decomposition approach for traffic control at isolated signalized intersections. *Probability in the Engineering and Informational Sciences* **22,** 587–602.

[114] HAJEK, B. AND ZHU, J. (2011). The missing piece syndrome in peer-to-peer communication. *Stochastic Systems* **1,** 246–273.

[115] HALFIN, S. (1972). Steady-state distribution for the buffer content of an M/G/1 queue with varying service rate. *SIAM Journal on Applied Mathematics* **23,** 356–363.

[116] HAQUE, L. AND ARMSTRONG, M. J. (2007). A survey of the machine interference problem. *European Journal of Operational Research* **179,** 469–482.

[117] HARKEMA, M., GIJSEN, B. M. M., VAN DER MEI, R. D. AND HOEKSTRA, Y. (2004). Middleware performance: A quantitative modeling approach. In *Proceedings of the International Symposium on Performance Evaluation of Computer and Communication Systems*. pp. 733–742.

[118] HARRIS, C. M. AND MARCHAL, W. G. (1988). State dependence in M/G/1 server-vacation models. *Operations Research* **36,** 560–565.

[119] HARRISON, J. M. AND WILLIAMS, R. J. (1987). Brownian models of open queueing networks with homogeneous customer populations. *Stochastics* **22,** 77–115.

[120] HARRISON, J. M. AND WILLIAMS, R. J. (1987). Multidimensional reflected Brownian motions having exponential stationary distributions. *The Annals of Probability* **15,** 115–137.

[121] HERNÁNDEZ-LERMA, O. AND LASSERRE, J. B. (1996). *Discrete-Time Markov Control Processes: Basic Optimality Criteria*. Springer, New York.

[122] HERZOG, U. AND ROLIA, J. A. (2001). Performance validation tools for software/hardware systems. *Performance Evaluation* **45,** 125–146.

[123] HIRAYAMA, T. (2009). Markovian polling systems: Functional computation for mean waiting times and its computational complexity. In *Advances in Queueing Theory and Network Applications*. Eds. W. Yue, Y. Takahashi, and H. Takagi. Springer, New York. pp. 119–146.

[124] HIRAYAMA, T. (2012). Analysis of multiclass Markovian polling systems with feedback and composite scheduling algorithms. *Annals of Operations Research* **198,** 83–123.

[125] HOLMA, H. AND TOSKALA, A. (2006). *HSDPA/HSUPA for UMTS: High Speed Radio Access for Mobile Communications*. John Wiley & Sons, Chichester.

[126] HOOGHIEMSTRA, G., KEANE, M. AND VAN DE REE, S. (1988). Power series for stationary distributions of coupled processor models. *SIAM Journal on Applied Mathematics* **48,** 1159–1166.

[127] HOPP, W. J., IRAVANI, S. M. R. AND YUEN, G. J. (2006). Operations systems with discretionary task completion. *Management Science* **53,** 61–77.

[128] ISRAR, T. A., LAU, D. H., FRANKS, R. G. AND WOODSIDE, C. M. (2005). Automatic generation of layered queuing software performance models from commonly available traces. In *Proceedings of the 5th International Workshop on Software and Performance*. pp. 147–158.

[129] IVANOVS, J., BOXMA, O. J. AND MANDJES, M. R. H. (2010). Singularities of the matrix exponent of a Markov additive process with one-sided jumps. *Stochastic Processes and their Applications* **120,** 1776–1794.

[130] JACKSON, J. R. (1957). Networks of waiting lines. *Operations Research* **5,** 518–521.

[131] JELENKOVIĆ, P. R., MOMČILOVIĆ, P. AND ZWART, B. (2004). Reduced load equivalence under subexponentiality. *Queueing Systems* **46,** 97–112.

[132] JONCKHEERE, M., VAN DER MEI, R. D. AND VAN DER WEIJ, W. (2010). Rate stability and output rates in queueing networks with shared resources. *Performance Evaluation* **67,** 28–42.

[133] JOSHI, K. R., HILTUNEN, M. A. AND JUNG, G. (2009). Performance aware regeneration in virtualized multitier applications. In *Proceedings of the 1st Workshop on Proactive Failure Avoidance, Recovery and Maintenance*.

[134] KÄHKIPURO, P. (1999). UML based performance modeling framework for object-oriented distributed systems. In *≪UML≫'99 - The Unified Modeling Language*. Eds. R. France and B. Rumpe. Vol. 1723 of *Lecture Notes in Computer Science*. Springer, Berlin/Heidelberg. pp. 356–371.

[135] KEILSON, J. AND SERVI, L. D. (1990). The distributional form of Little's law and the Fuhrmann-Cooper decomposition. *Operations Research Letters* **9,** 239–247.

[136] KELLA, O. AND WHITT, W. (1990). Diffusion approximations for queues with server vacations. *Advances in Applied Probability* **22,** 706–729.

[137] KELLY, F. P. (1975). Networks of queues with customers of different types. *Journal of Applied Probability* **12,** 542–554.

[138] KELLY, F. P. (1979). *Reversibility and Stochastic Networks*. Wiley, Chichester.

[139] KHALIL, H. K. (2002). *Nonlinear Systems*. Prentice Hall, Englewood Cliffs.

[140] KINGMAN, J. F. C. (1961). The single server queue in heavy traffic. *Mathematical Proceedings of the Cambridge Philosophical Society* **57,** 902–904.

[141] KINGMAN, J. F. C. (1965). The heavy traffic approximation in the theory of queues. In *Proceedings of the Symposium on Congestion Theory*. pp. 137–159.

[142] KINO, I. (1997). Two-layer queueing networks. *Journal of the Operations Research* **40,** 163–204.

[143] KLEINROCK, L. (1976). *Queueing Systems, Volume II: Computer Applications*. Wiley, New York.

[144] KLEINROCK, L. AND LEVY, H. (1988). The analysis of random polling systems. *Operations Research* **36,** 716–732.

[145] KONHEIM, A. G., LEVY, H. AND SRINIVASAN, M. M. (1994). Descendant set: An efficient approach for the analysis of polling systems. *IEEE Transactions on Communications* **42,** 1245–1253.

[146] KOOLE, G. M. (1994). On the power series algorithm. In *Performance Evaluation of Parallel and Distributed Systems - Solution Methods*. Eds. O. J. Boxma and G. M. Koole. CWI, Amsterdam. pp. 139–155. CWI Tract 105 & 106.

[147] KOOLE, G. M. (2006). Monotonicity in Markov reward and decision chains: Theory and applications. *Foundations and Trends in Stochastic Systems* **1,** 1–76.

[148] KOOLE, G. M. AND MANDELBAUM, A. (2002). Queueing models of call centers: An introduction. *Annals of Operations Research* **113,** 41–59.

[149] KOSIŃSKI, K. M., BOXMA, O. J. AND ZWART, B. (2011). Convergence of the all-time supremum of a Lévy process in the heavy-traffic regime. *Queueing Systems* **67,** 295–304.

[150] KOZIOLEK, H. AND REUSSNER, R. (2008). A model transformation from the Palladio component model to layered queueing networks. In *Performance Evaluation: Metrics, Models and Benchmarks*. Eds. S. Kounev, I. Gorton, and K. Sachs. Vol. 5119 of *Lecture Notes in Computer Science*. Springer, Berlin/Heidelberg. pp. 58–78.

[151] KRISHNAMURTHY, D., ROLIA, J. A. AND XU, M. (2011). WAM – The Weighted Average Method for predicting the performance of systems with bursts of customer sessions. *IEEE Transactions on Software Engineering* **37,** 718–735.

[152] KURASUGI, T. AND KINO, I. (1999). Approximation methods for two-layer queueing models. *Performance Evaluation* **36–37,** 55–70.

[153] KUSHNER, H. J. AND YIN, G. G. (2003). *Stochastic Approximation and Recursive Algorithms and Applications*. Springer, New York.

[154] KYPRIANOU, A. E. (2006). *Introductory Lectures on Fluctuations of Lévy Processes with Applications*. Springer, Berlin/Heidelberg.

[155] LANGARIS, C. (1999). Markovian polling systems with mixed service disciplines and retrial customers. *TOP* **7,** 305–322.

[156] LAVENBERG, S. S. AND REISER, M. (1980). Stationary state probabilities at arrival instants for closed queueing networks with multiple types of customers. *Journal of Applied Probability* **17,** 1048–1061.

[157] LEVY, H. (1989). Analysis of cyclic polling systems with binomial gated service. In *Performance of Distributed and Parallel Systems*. Eds. T. Hasegawa, H. Takagi, and Y. Takahashi. North-Holland, Amsterdam. pp. 127–139.

[158] LEVY, H. AND SIDI, M. (1990). Polling systems: Applications, modeling and optimization. *IEEE Transactions on Communications* **38,** 1750–1760.

[159] LI, J. Z., CHINNECK, J., WOODSIDE, C. M. AND LITOIU, M. (2009). Fast scalable optimization to configure service systems having cost and quality of service constraints. In *Proceedings of the 6th International Conference on Autonomic Computing*. pp. 159–168.

[160] LI, L. J. AND FRANKS, R. G. (2009). Performance modeling of systems using fair share scheduling with layered queueing networks. In *Proceedings of the 17th IEEE/ACM International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems*. pp. 1–10.

[161] LINDLEY, D. V. (1952). The theory of queues with a single server. **48,** 277–289.

[162] LIPPMAN, S. A. (1975). Applying a new device in the optimization of exponential queuing systems. *Operations Research* **23,** 687–710.

[163] LIPPMAN, S. A. (1975). On dynamic programming with unbounded rewards. *Management Science* **21,** 1225–1233.

[164] LITOIU, M., KRISHNAMURTHY, D. AND ROLIA, J. A. (2002). Performance stress vectors and capacity planning for e-commerce applications. *International Journal on Digital Libraries* **3,** 309–315.

[165] LITOIU, M. AND ROLIA, J. A. (2000). Object allocation for distributed applications with complex workloads. In *Proceedings of the 11th Conference on Computer Performance Evaluation, Modelling Techniques and Tools*. pp. 25–39.

[166] LITOIU, M., ROLIA, J. A. AND SERAZZI, G. (2000). Designing process replication and activation: A quantitative approach. *IEEE Transactions on Software Engineering* **26,** 1168–1178.

[167] LITVAK, N. AND VLASIOU, M. (2010). A survey on performance analysis of warehouse carousel systems. *Statistica Neerlandica* **64,** 401–447.

[168] LIU, T., BEHROOZI, A. AND KUMARAN, S. (2003). A performance model for a business process integration middleware. In *Proceedings of the IEEE International Conference on Electronic Commerce*. pp. 191–198.

[169] LIU, T., KUMARAN, S. AND CHUNG, J. (2004). Performance engineering of a Java-based e-commerce system. In *Proceedings of the 2004 IEEE International Conference on e-Technology, e-Commerce and e-Service*. pp. 33–37.

[170] LIU, Z., NAIN, P. AND TOWSLEY, D. (1992). On optimal polling policies. *Queueing Systems* **11,** 59–83.

[171] 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* **19,** 173–178.

[172] MACK, C., MURPHY, T. AND 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* **19,** 166–172.

[173] MAHABHASHYAM, S. R. AND GAUTAM, N. (2005). On queues with Markov modulated service rates. *Queueing Systems* **51,** 89–113.

[174] MARTENS, A., KOZIOLEK, H., BECKER, S. AND REUSSNER, R. (2010). Automatically improve software architecture models for performance, reliability, and cost using evolutionary algorithms. In *Proceedings of the 1st joint WOSP/SIPEW international conference on Performance engineering*. pp. 105–116.

[175] MCGINNIS, L. F., HAN, M. H. AND WHITE, J. A. (1986). Analysis of rotary rack operations. In *Proceedings of the 7th International Conference on Automation in Warehousing*. pp. 165–171.

[176] MENASCÉ, D. (2002). Two-level iterative queuing modeling of software contention. In *Proceedings of the 10th International Workshop on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems*. pp. 267–276.

[177] MITRA, D. (1988). Stochastic theory of a fluid model of producers and consumers coupled by a buffer. *Advances in Applied Probability* **20,** 646–676.

[178] MROZ, M. AND FRANKS, R. G. (2009). A performance experiment system supporting fast mapping of system issues. In *Proceedings of the 4th International ICST Conference on Performance Evaluation Methodologies and Tools*.

[179] NAIR, J., WIERMAN, A. AND ZWART, B. (2010). Tail-robust scheduling via limited processor sharing. *Performance Evaluation* **67,** 978–995.

[180] NEUTS, M. F. (1981). *Matrix-Geometric Solutions in Stochastic Models: An Algorithmic Approach*. Johns Hopkins University Press, Baltimore.

[181] NORMAN, J. M. (1972). *Heuristic Procedures in Dynamic Programming*. Manchester University Press, Manchester.

[182] NÚÑEZ-QUEIJA, R. (1998). A queueing model with varying service rate for ABR. In *Computer Performance Evaluation: Modelling Techniques and Tools*. Eds. R. Puigjaner, N. N. Savino, and B. Serra. Vol. 1469 of *Lecture Notes in Computer Science*. Springer, Berlin/Heidelberg. pp. 93–104.

[183] NUYENS, M. AND VAN DER WEIJ, W. (2007). Monotonicity in the limited processor-sharing queue. *Stochastic Models* **25,** 408–419.

[184] OLSEN, T. L. AND VAN DER MEI, R. D. (2003). Polling systems with periodic server routeing in heavy traffic: Distribution of the delay. *Journal of Applied Probability* **40,** 305–326.

[185] OTT, T. J. AND KRISHNAN, K. R. (1992). Separable routing: A scheme for state-dependent routing of circuit switched telephone traffic. *Annals of Operations Research* **35,** 43–68.

[186] PACIFICI, G., SEGMULLER, W., SPREITZER, M. AND TANTAWI, A. N. (2008). CPU demand for web serving: Measurement analysis and dynamic estimation. *Performance Evaluation* **65,** 531–553.

[187] PAGANINI, F., FERRAGUT, A. AND ZUBELDIA, M. (2013). Dynamics of heterogeneous peer-to-peer networks. In *Proceedings of the 52nd IEEE Conference on Decision and Control*. pp. 3293–3298.

[188] PARK, B. C., PARK, J. Y. AND FOLEY, R. D. (2003). Carousel system performance. *Journal of Applied Probability* **40,** 602–612.

[189] PEREL, E. AND YECHIALI, U. (2008). Queues where customers of one queue act as servers of the other queue. *Queueing Systems* **60,** 271–288.

[190] PEREL, E. AND YECHIALI, U. (2013). On customers acting as servers. *Asia-Pacific Journal of Operational Research* **30,** 1–23.

[191] PETRIU, D. B., AMYOT, D. AND WOODSIDE, C. M. (2003). Scenario-based performance engineering with UCMNav. In *SDL 2003: System Design*. Eds. R. Reed and J. Reed. Vol. 2708 of *Lecture Notes in Computer Science*. Springer, Berlin/Heidelberg. pp. 18–35.

[192] PETRIU, D. B. AND WOODSIDE, C. M. (2002). Analysing software requirements specifications for performance. In *Proceedings of the 3rd International Workshop on Software and Performance*. pp. 1–9.

[193] PETRIU, D. B. AND WOODSIDE, C. M. (2005). Software performance models from system scenarios. *Performance Evaluation* **61,** 65–89.

[194] PETRIU, D. C., AMER, H., MAJUMDAR, S. AND ABDULL-FATAH, I. (2000). Using analytic models predicting middleware performance. In *Proceedings of the 2nd International Workshop on Software and Performance*. pp. 189–194.

[195] PETRIU, D. C. AND SHEN, H. (2002). Applying the UML performance profile: Graph grammar-based derivation of LQN models from UML specifications. In *Computer*

Performance Evaluation: Modelling Techniques and Tools. Eds. T. Field, P. G. Harrison, J. Bradley, and U. Harder. Vol. 2324 of *Lecture Notes in Computer Science*. Springer, Berlin/Heidelberg. pp. 159–177.

[196] PETRIU, D. C., SHEN, H. AND SABETTA, A. (2007). Performance analysis of aspect-oriented UML models. *Software & Systems Modeling* **6,** 453–471.

[197] PETRIU, D. C. AND WANG, X. (2000). Deriving software performance models from architectural patterns by graph transformations. In *Theory and Application of Graph Transformations*. Eds. H. Ehrig, G. Engels, H. Kreowski, and G. Rozenberg. Vol. 1764 of *Lecture Notes in Computer Science*. Springer, Berlin/Heidelberg. pp. 475–488.

[198] PETRIU, D. C. AND WANG, X. (2000). From UML descriptions of high-level software architectures to LQN performance models. In *Applications of Graph Transformations with Industrial Relevance*. Eds. M. Nagl, A. Schürr, and M. Münch. Vol. 1779 of *Lecture Notes in Computer Science*. Springer, Berlin/Heidelberg. pp. 47–63.

[199] PETRIU, D. C. AND WOODSIDE, C. M. (2002). Software performance models from system scenarios in Use Case Maps. In *Computer Performance Evaluation: Modelling Techniques and Tools*. Eds. T. Field, P. G. Harrison, J. Bradley, and U. Harder. Vol. 2324 of *Lecture Notes in Computer Science*. Springer, Berlin/Heidelberg. pp. 141–158.

[200] PETRIU, D. C., WOODSIDE, C. M., PETRIU, D. B., XU, J., ISRAR, T. A., GEORG, G., FRANCE, R., BIEMAN, J. M., HOUMB, S. H. AND JÜRJENS, J. (2007). Performance analysis of security aspects in UML models. In *Proceedings of the 6th International workshop on Software and Performance*. pp. 91–102.

[201] PURDUE, P. (1974). The M/M/1 queue in a Markovian environment. *Operations Research* **22,** 562–569.

[202] PUTERMAN, M. L. (1994). *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Chichester.

[203] QIANG, Y., LI, Y. AND CHEN, J. (2009). The workload adaptation in autonomic DBMSs based on layered queuing network model. In *Proceedings of the 2nd International Workshop on Knowledge Discovery and Data Mining*. pp. 781–785.

[204] RAMESH, S. AND PERROS, H. G. (2000). A multilayer client-server queueing network model with synchronous and asynchronous messages. *IEEE Transactions on Software Engineering* 1086–1100.

[205] REIMAN, M. I. (1984). Some diffusion approximations with state space collapse. In *Modelling and Performance Evaluation Methodology*. Eds. F. Baccelli and G. Fayolle. Vol. 60 of *Lecture Notes in Control and Information Sciences*. Springer, Berlin/Heidelberg. pp. 209–240.

[206] REIMAN, M. I. AND SIMON, B. (1988). An interpolation approximation for queueing systems with Poisson input. *Operations Research* **36,** 454–469.

[207] REISER, M. AND LAVENBERG, S. S. (1980). Mean-value analysis of closed multichain queuing networks. *Journal of the ACM* **27,** 313–322.

[208] RESING, J. A. C. (1993). Polling systems and multitype branching processes. *Queueing Systems* **13,** 409–426.

[209] REVUZ, D. AND YOR, M. (1999). *Continuous Martingales and Brownian Motion*. Springer, Berlin/Heidelberg.

[210] RISSE, T., ABERER, K., WOMBACHER, A., SURRIDGE, M. AND TAYLOR, S. (2004). Configuration of distributed message converter systems. *Performance Evaluation* **58,** 43–80.

[211] ROLIA, J. A. (1988). Performance estimates for systems with software servers: The lazy boss method. In *Proceedings of the 7th SCCC International Conference On Computer Science*. pp. 25–43.

[212] ROLIA, J. A., CASALE, G., KRISHNAMURTHY, D., DAWSON, S. AND KRAFT, S. (2009). Predictive modelling of SAP ERP applications: Challenges and solutions. In *Proceedings of the 4th International ICST Conference on Performance Evaluation Methodologies and Tools*.

[213] ROLIA, J. A., KALBASI, A., KRISHNAMURTHY, D. AND DAWSON, S. (2010). Resource demand modeling for multi-tier services. In *Proceedings of the 1st Joint WOSP/SIPEW International Conference on Performance Engineering*. pp. 207–216.

[214] ROLIA, J. A. AND SEVCIK, K. C. (1995). The method of layers. *IEEE Transactions on Software Engineering* **21,** 689–700.

[215] ROLIA, J. A. AND VETLAND, V. (1995). Parameter estimation for performance models of distributed application systems. In *Proceedings of the 1995 Conference of the Centre for Advanced Studies on Collaborative Research*.

[216] RUMBAUGH, J., JACOBSON, I. AND BOOCH, G. (2004). *The Unified Modeling Language Reference Manual, Second Edition*. Addison Wesley, Boston.

[217] SANGHAVI, S., HAJEK, B. AND MASSOULIÉ, L. (2007). Gossiping with multiple messages. *IEEE Transactions on Information Theory* **53,** 4640–4654.

[218] SASSEN, S. A. E., TIJMS, H. C. AND NOBEL, R. D. (1997). A heuristic rule for routing customers to parallel servers. *Statistica Neerlandica* **51,** 107–121.

[219] SCHWEITZER, P. J. (1979). Approximate analysis of multiclass closed networks of queues. In *Proceedings of the International Conference on Stochastic Control and Optimization*. pp. 25–29.

[220] SERFOZO, R. F. (1999). *Introduction to Stochastic Networks*. Springer, New York.

[221] SEVCIK, K. C. AND MITRANI, I. (1981). The distribution of queueing network states at input and output instants. *Journal of the Association for Computing Machinery* **28,** 358–371.

[222] SHEIKH, F. AND WOODSIDE, C. M. (1997). Layered analytic performance modelling of a distributed database system. In *Proceedings of the 17th International Conference on Distributed Computing Systems*. pp. 482–490.

[223] SHNEER, S. AND WACHTEL, V. (2011). A unified approach to the heavy-traffic analysis of the maximum of random walks. *Theory of Probability and Its Applications* **55,** 332–341.

[224] SHOUSHA, C., PETRIU, D. C., JALNAPURKAR, A. AND NGO, K. (1998). Applying performance modelling to a telecommunication system. In *Proceedings of the 1st International Workshop on Software and Performance*. pp. 1–6.

[225] SIEBERT, F. (1999). Real-time garbage collection in multi-threaded systems on a single processor. In *Proceedings of the 20th IEEE Real-Time Systems Symposium*. pp. 277–278.

[226] SMITH, C. U. (1990). *Performance Engineering of Software Systems*. Addison Wesley, Reading.

[227] SRINIVASAN, M. M. (1991). Nondeterministic polling systems. *Management Science* **37,** 667–681.

[228] STECKE, K. E. AND ARONSON, J. E. (1985). Review of operator/machine interference models. *International Journal of Production Research* **23,** 129–151.

[229] STEICHEN, J. L. (2001). A functional central limit theorem for Markov additive processes with an application to the closed Lu-Kumar network. *Stochastic Models* **17,** 459–489.

[230] STIDHAM JR., S. AND WEBER, R. R. (1989). Monotonic and insensitive optimal policies for control of queues with undiscounted costs. *Operations Research* **37,** 611–625.

[231] SZEKLI, R. (1995). *Stochastic Ordering and Dependence in Applied Probability*. Springer, New York.

[232] TAKÁCS, L. (1962). *Introduction to the Theory of Queues*. Oxford University Press, New York.

[233] TAKAGI, H. (1986). *Analysis of Polling Systems*. MIT Press, Cambridge.

[234] TAKINE, T. (2005). Single-server queues with Markov-modulated arrivals and service speed. *Queueing Systems* **49,** 7–22.

[235] TANENBAUM, A. S. AND WETHERALL, D. J. (2010). *Computer Networks, Fifth Edition*. Prentice Hall, Englewood Cliffs.

[236] TAWHID, R. AND PETRIU, D. C. (2008). Towards automatic derivation of a product performance model from a UML software product line model. In *Proceedings of the 7th International Workshop on Software and Performance*. pp. 91–102.

[237] TERTILT, D. AND KRCMAR, H. (2011). Generic performance prediction for ERP and SOA applications. In *Proceedings of the 12th European Conference on Information Systems*.

[238] TIJMS, H. C. (1994). *Stochastic Models: An Algorithmic Approach*. John Wiley & Sons, Chichester.

[239] TITCHMARSH, E. C. (1939). *Theory of Functions*. Oxford University Press, London.

[240] TIWARI, N. AND MYNAMPATI, P. (2006). Experiences of using LQN and QPN tools for performance modeling of a J2EE application. In *Proceedings of the 32nd International Computer Measurement Group Conference*. pp. 537–548.

[241] TRIBASTONE, M. (2010). Relating layered queueing networks and process algebra models. In *Proceedings of the 1st Joint WOSP/SIPEW International Conference on Performance Engineering*. pp. 183–194.

[242] TRIBASTONE, M. (2013). A fluid model for layered queueing networks. *IEEE Transactions on Software Engineering* **39,** 744–756.

[243] TRIBASTONE, M., MAYER, P. AND WIRSING, M. (2010). Performance prediction of service-oriented systems with layered queueing networks. In *Leveraging Applications of Formal Methods, Verification, and Validation*. Eds. T. Margaria and B. Steffen. Vol. 6416 of *Lecture Notes in Computer Science*. Springer, Berlin/Heidelberg. pp. 51–65.

[244] TSE, D. AND VISWANATH, P. (2005). *Fundamentals of Wireless Communication*. Cambridge University Press, Cambridge.

[245] TZENOVA, E. I., ADAN, I. J. B. F. AND KULKARNI, V. G. (2005). Fluid models with jumps. *Stochastic Models* **21,** 37–55.

[246] UFIMTSEV, A. AND MURPHY, L. (2006). Performance modeling of a JavaEE component application using layered queuing networks: Revised approach and a case study. In *Proceedings of the 2006 Conference on Specification and Verification of Component-Based Systems*. pp. 11–18.

[247] VAN DEN HOUT, W. B. AND BLANC, J. P. C. (1995). Development and justification of the power-series algorithm for BMAP-systems. *Communications in Statistics. Stochastic Models* **11,** 471–496.

[248] VAN DER MEI, R. D. (1999). Distribution of the delay in polling systems in heavy traffic. *Performance Evaluation* **38,** 133–148.

[249] VAN DER MEI, R. D. (2007). Towards a unifying theory on branching-type polling systems in heavy traffic. *Queueing Systems* **57,** 29–46.

[250] VAN DER MEI, R. D., GIJSEN, B. M. M. AND MOHY EL DINE, S. (2004). Throughput optimisation in a two-layered tandem of multi-server queues. In *Proceedings of the 2nd International Working Conference on Performance Modelling and Evaluation of Heterogeneous Networks*.

[251] VAN DER MEI, R. D., HARIHARAN, R. AND REESER, P. K. (2001). Web server performance modeling. *Telecommunication Systems* **16,** 361–378.

[252] VAN DER MEI, R. D. AND WINANDS, E. M. M. (2008). Heavy traffic analysis of polling models by mean value analysis. *Performance Evaluation* **65,** 400–416.

[253] VAN DER WEIJ, W., BHULAI, S. AND VAN DER MEI, R. D. (2009). Dynamic thread assignment in web server performance optimization. *Performance Evaluation* **66,** 301–310.

[254] VAN DER WEIJ, W. AND VAN DER MEI, R. D. (2005). Stability and throughput in a two-layered network of multi-server queues. In *Proceedings of the 3rd International Working Conference on Performance Modelling and Evaluation of Heterogeneous Networks*.

[255] VAN DER WEIJ, W., VAN DER MEI, R. D., GIJSEN, B. M. M. AND PHILLIPSON, F. (2005). Optimal server assignment in a two-layered tandem of multi-server queues. In *Proceedings of the 3rd International Working Conference on Performance Modelling and Evaluation of Heterogeneous Networks*.

[256] VAN DER WEIJ, W., VAN DIJK, N. M. AND VAN DER MEI, R. D. (2012). Product-form results for two-station networks with shared resources. *Performance Evaluation* **69,** 662–683.

[257] VAN DOORN, E. A. AND REGTERSCHOT, G. J. K. (1988). Conditional PASTA. *Operations Research Letters* **7,** 229–232.

[258] VAN HOECKE, S., VERDICKT, T., DE TURCK, F., DHOEDT, B. AND DEMEESTER, P. (2005). Modeling the performance of the web service platform using layered queueing networks. In *Proceedings of the International Conference on Software Engineering Research and Practice*. pp. 627–633.

[259] VAN VUUREN, M. AND WINANDS, E. M. M. (2007). Iterative approximation of k-limited polling systems. *Queueing Systems* **55,** 161–178.

[260] VAN WIJK, A. C. C., ADAN, I. J. B. F., BOXMA, O. J. AND WIERMAN, A. (2012). Fairness and efficiency for polling models with the κ-gated service discipline. *Performance Evaluation* **69,** 274–288.

[261] VANGHI, V., DAMNJANOVIC, A. AND VOJCIC, B. (2004). *The cdma2000 System for Mobile Communications: 3G Wireless Evolution*. Prentice Hall, Englewood Cliffs.

[262] VERDICKT, T., DHOEDT, B., GIELEN, F. AND DEMEESTER, P. (2003). Modelling the performance of CORBA using layered queueing networks. In *Proceedings of the 29th EUROMICRO Conference*. pp. 117–123.

[263] VISHNEVSKII, V. M. AND SEMENOVA, O. M. (2006). Mathematical models to study the polling systems. *Automation and Remote Control* **67,** 173–220.

[264] VLASIOU, M. (2006). Lindley-Type Recursions. *PhD thesis*. Eindhoven University of Technology, Eindhoven, The Netherlands.

[265] VLASIOU, M. (2007). A non-increasing Lindley-type equation. *Queueing Systems* **56,** 41–52.

[266] VLASIOU, M. AND ADAN, I. J. B. F. (2005). An alternating service problem. *Probability in the Engineering and Informational Sciences* **19,** 409–426.

[267] VLASIOU, M., ADAN, I. J. B. F. AND BOXMA, O. J. (2009). A two-station queue with dependent preparation and service times. *European Journal of Operational Research* **195,** 104–116.

[268] VLASIOU, M., ZHANG, J., ZWART, B. AND VAN DER MEI, R. D. (2012). Separation of timescales in a two-layered network. In *Proceedings of the 24th International Teletraffic Congress*.

[269] WARTENHORST, P. (1995). *N* parallel queueing systems with server breakdown and repair. *European Journal of Operational Research* **82,** 302–322.

[270] WEBER, R. R. AND STIDHAM JR., S. (1987). Optimal control of service rates in networks of queues. *Advances in Applied Probability* **19,** 202–218.

[271] WESTSTRATE, J. A. (1992). Analysis and Optimization of Polling Models. *PhD thesis*. Katholieke Universiteit Brabant, Tilburg, The Netherlands.

[272] WESTSTRATE, J. A. AND VAN DER MEI, R. D. (1994). Waiting times in a two-queue model with exhaustive and Bernoulli service. *Zeitschrift für Operations Research* **40,** 289–303.

[273] WHITT, W. (1989). An interpolation approximation for the mean workload in a GI/G/1 queue. *Operations Research* **37,** 936–952.

[274] WHITT, W. (1992). Asymptotic formulas for Markov processes with applications to simulation. *Operations Research* **40,** 279–291.

[275] WHITT, W. (2002). *Stochastic-Process Limits*. Springer, New York.

[276] WIJNGAARD, J. (1979). Decomposition for dynamic programming in production and inventory control. *Engineering and Process Economics* **4,** 385–388.

[277] WILLIAMS, D. (1991). *Probability with Martingales*. Cambridge University Press, Cambridge.

[278] WOODSIDE, C. M. (1989). Throughput calculation for basic stochastic rendezvous networks. *Performance Evaluation* **9,** 143–160.

[279] WOODSIDE, C. M., NEILSON, J. E., PETRIU, D. C. AND MAJUMDAR, S. (1995). The stochastic rendezvous network model for performance of synchronous client-server-like distributed software. *IEEE Transactions on Computers* **44,** 20–34.

[280] WOODSIDE, C. M., NERON, E., HO, E. AND MONDOUX, B. (1986). An "active server" model for the performance of parallel programs written using rendezvous. *Journal of Systems and Software* **6,** 125–131.

[281] WOODSIDE, C. M., PETRIU, D. C., PETRIU, D. B., XU, J., ISRAR, T. A., GEORG, G., FRANCE, R., BIEMAN, J. M., HOUMB, S. H. AND JÜRJENS, J. (2009). Performance analysis of security aspects by weaving scenarios extracted from UML models. *Journal of Systems and Software* **82,** 56–74.

[282] WOODSIDE, C. M., ZHENG, T. AND LITOIU, M. (2005). The use of optimal filters to track parameters of performance models. In *Proceedings of the 2nd International Conference on the Quantitative Evaluation of Systems*. pp. 74–83.

[283] WOODSIDE, C. M., ZHENG, T. AND LITOIU, M. (2006). Service system resource management based on a tracked layered performance model. In *Proceedings of the 3rd International Conference on Autonomic Computing*. pp. 175–184.

[284] XU, J. (2012). Rule-based automatic software performance diagnosis and improvement. *Performance Evaluation* **69,** 525–550.

[285] XU, J., UFIMTSEV, A., WOODSIDE, C. M. AND MURPHY, L. (2006). Performance modeling and prediction of Enterprise JavaBeans with layered queuing network templates. In *Proceedings of the 2005 Conference on Specification and Verification of Component-Based Systems*.

[286] XU, J. AND WOODSIDE, C. M. (2005). Template-driven performance modeling of Enterprise Java Beans. In *Proceedings of the Workshop on Middleware for Web Services*. pp. 57–64.

[287] ZHANG, J., DAI, J. G. AND ZWART, B. (2009). Law of large number limits of limited processor-sharing queues. *Mathematics of Operations Research* **34,** 937–970.

[288] ZHANG, J., DAI, J. G. AND ZWART, B. (2011). Diffusion limits of limited processor sharing queues. *The Annals of Applied Probability* **21,** 745–799.

[289] ZHANG, J. AND ZWART, B. (2008). Steady state approximations of limited processor sharing queues in heavy traffic. *Queueing Systems* **60,** 227–246.

[290] ZHANG, Q., CHERKASOVA, L. AND SMIRNI, E. (2007). A regression-based analytic model for dynamic resource provisioning of multi-tier applications. In *Proceedings of the 4th International Conference on Autonomic Computing*.

[291] ZHENG, T. AND WOODSIDE, C. M. (2003). Heuristic optimization of scheduling and allocation for distributed systems with soft deadlines. In *Computer Performance Evaluation: Modelling Techniques and Tools*. Eds. P. Kemper and W. H. Sanders. Vol. 2794 of *Lecture Notes in Computer Science*. Springer, Berlin/Heidelberg. pp. 169–181.

[292] ZHENG, T. AND WOODSIDE, C. M. (2005). Fast estimation of probabilities of soft deadline misses in layered software performance models. In *Proceedings of the 5th International Workshop on Software and Performance*. pp. 181–186.

[293] ZHENG, T., YANG, J., WOODSIDE, C. M., LITOIU, M. AND ISZLAI, G. (2005). Tracking time-varying parameters in software systems with extended kalman filters. In *Proceedings of the 2005 Conference of the Centre for Advanced Studies on Collaborative Research*. pp. 334–345.

[294] ZHU, J. AND HAJEK, B. (2012). Stability of a peer-to-peer communication system. *IEEE Transactions on Information Theory* **58,** 4693–4713.

[295] ZWART, B. (2004). Heavy-traffic asymptotics for the single-server queue with random order of service. *Operations Research Letters* **33,** 511–518.

SUMMARY

Layered Queueing Networks – Performance Modelling, Analysis and Optimisation

This dissertation is concerned with the mathematical study of layered queueing networks. This topic can be placed within the domain of queueing theory, which analyses congestion phenomena and provides methods to evaluate the performance of complex systems arising in areas such as computer and communication networks, supply chains, traffic networks, manufacturing and customer contact centers. Typical models involve servers working on customers that arrive randomly and require a random amount of service.

Recent applications in engineering, business and the public sector led to systems with much more complex, often layered, service architectures, where entities that provide service at one layer can request service at a lower layer. This naturally leads to the modelling of such applications as layered queueing networks. These queueing networks consist of multiple layers and have the distinctive property that servers of any layer can act as customers in the layer directly below. Mathematical analysis of this subclass of networks is very challenging, since the resulting interactions between layers must be taken into account. For instance, the performance of lower-layer servers may heavily impact the congestion levels incurred by higher-layer customers. In this thesis, we perform an in-depth analysis of three such layered queueing networks consisting of two layers, where the interactions between the layers cannot be ignored. With this analysis, we aim to gain insights into the impact of the layer interactions on the performance and control of the queueing networks considered and layered queueing networks in general. Furthermore, the methods used and the analysis performed might be used as a starting point to study queueing networks with a larger number of layers.

The first network, which we study in Chapters 2–6, is an extension of what is known in queueing theory as the machine repair model. This model consists of a number of machines working in parallel in a manufacturing setting and a single repairman. As soon as a machine fails, it joins a repair queue in order to be repaired by the repairman. Thus, the machines are customers of the repairman, who is the server. In practice, however, a machine also acts as a server in a higher layer when it processes products. We therefore extend the machine repair model by adding queues of products to the model. Because of the dual role of the machines in different layers, this model constitutes a layered queueing network. For this extended model, we obtain several approximations for the waiting time of the products, while explicitly taking the characteristics of the repairman into account. We do so by deriving light-traffic and heavy-traffic asymptotics of the extended model in Chapters 2 and 3, respectively, which we combine to form highly accurate approximations for the mean queue lengths of the queues of products in Chapter 4. In Chapter 5, we derive an accurate approximation for the complete queue length distribution by carefully

studying the dependence structure between the layers. From the results of these chapters, it is apparent that the characteristics of the repairman have a large impact on the delay incurred by the products as a result of machine failures. Therefore, in Chapter 6, by utilising the framework of Markov decision processes, we formulate an answer to the question of how the repairman should allocate his repair resources to the machines in order to minimise the delay incurred at each of the machines.

Chapters 7–9 are devoted to the study of the second layered queueing network, which involves a queueing network consisting of multiple queues attended by a single server. The server visits the queues in some order to render service to the customers waiting at each of the queues and incurs stochastic switch-over times when he moves from one queue to another. The order in which the server visits the queues is assumed to be determined by an external random environment. More specifically, we assume that this order is governed by a discrete-time Markov chain. We study this model with a view towards an application to wireless random-access networks, where nodes share a medium (i.e. the server) to transmit packets waiting in packet buffers (i.e. the queues). This queueing network evidently falls in the class of layered queueing networks. The nodes are servers in their role of packet transmitter, but they can also be interpreted as customers in a lower layer, since they incur delays in claiming the medium to execute their transmitted tasks. In Chapters 7 and 8, we perform an in-depth analysis of the waiting times of the first-layer customers in a variety of settings, while taking the routing dynamics of the second-layer server into account. The results obtained in these chapters, which we believe to be of independent interest, also serve as building blocks for Chapter 9. In this chapter, we formulate a distributed algorithm to optimise these waiting times in the setting of wireless random-access networks, where the nodes typically suffer the problem of incomplete information due to the decentralised nature of these networks.

Finally, Chapters 10 and 11 concern themselves with the third layered queueing network considered in this thesis, where customers first undergo a preparation phase at a service station and subsequently require a phase of service from a specialised server who polls the service stations. This problem originates from warehousing, but also has applications in healthcare, where surgeons poll multiple surgery rooms. As the service stations act as both customers and servers in different layers (they provide a phase of preparation, but are blocked when this phase ends and the server is not available), this third model also constitutes a layered queueing network. Observe that the specialised server, however, also has a dual role, since the server has to wait at times for a preparation phase of a customer to finish. In these cases, the server becomes a customer in some sense. In Chapter 10, under the assumption of an infinite number of waiting customers and cyclic routing of the specialised server through the service stations, we provide a detailed analysis of the waiting times incurred by the server. In doing so, we identify several parameter effects that influence this waiting time. In Chapter 11, we extensively investigate the effects of the removal of the restriction of cyclic routing by the server, which turn out to be major.

CURRICULUM VITAE

Jan-Pieter Dorsman was born in Amstelveen, The Netherlands, on March 31, 1987. In 2005, he finished secondary education at the St. Ignatiusgymnasium in Amsterdam, after which he went to study Business Mathematics and Informatics at the VU University Amsterdam. For several years during this period, Jan-Pieter acted as a teaching assistant for a variety of courses in mathematics at the same university. He received his Master's degree (cum laude) in 2010, following an internship project on the performance analysis and optimisation of polling systems at the Centrum Wiskunde & Informatica (CWI) in Amsterdam.

Directly after obtaining his MSc degree, Jan-Pieter started a PhD project at the Eindhoven University of Technology and CWI. Besides a substantial number of teaching duties, he conducted research on the performance modelling, analysis and optimisation of layered queueing models under the supervision of Onno Boxma, Rob van der Mei and Maria Vlasiou. The majority of the results of this study has been compiled in this thesis. Furthermore, the research conducted has led to a number of publications in international journals and conference proceedings. For one of these publications, he received a best paper award at the International Conference on Operations Research and Enterprise Systems (ICORES) in 2013.

During the last year of his PhD project, Jan-Pieter spent a few months at the department of Telecommunications and Information Processing (TELIN) at Ghent University in Belgium. Although he intends to continue his academic activities, his PhD project ends with the realisation of this thesis, which Jan-Pieter defends on February 17, 2015.