Scheduling Policies in Service Networks

by

Jianfu Wang

A thesis submitted in conformity with the requirements
for the degree of Doctor of Philosophy
Graduate Department of Rotman School of Management
University of Toronto

Abstract

Scheduling Policies in Service Networks

Jianfu Wang

Doctor of Philosophy

Graduate Department of Rotman School of Management

University of Toronto

2014

In this thesis, we study different scheduling policies in service networks. In Chapter 2, we consider two service level (SL) measures in a two-server tandem queue system: the average sojourn time and the probability of long waits. We demonstrate that a family of Threshold Based Policies (TBP) can reduce the probability of long waits while maintaining sojourn times that are only slightly higher than those of a non-idling policy. In Chapter 3, we present a case study for improving the operations of a healthcare provider that has an open-shop queueing network. We propose an effective implementation of Dynamic Scheduling Policies (DSPs) and a generalized TBP to improve the SL in an open-shop queueing networks. Using a simulation model we demonstrate that an open-shop queueing network can be managed in a systematic fashion to deliver improved SL. In Chapter 4, we study the waiting time distribution of two different priority classes in an M/M/c queue with different service times. For the $c = 2$ case, we provide closed form expression of the Generating Function (GF) of the number of low-priority jobs in the system, which can lead to the waiting time distribution. For $c > 2$ case, we present an efficient numerical algorithm for deriving this GF. We discuss several insights gained from numerical results. Both Chapter 2 and 3 were supervised by Professors Opher Baron, Oded Berman, and Dmitry Krass. Chapter 4 was supervised by Professors Opher Baron and Alan Scheller-Wolf.

Acknowledgements

This thesis would not have been possible without the support of many people. I wish to express my gratitude to my supervisors, Professors Opher Baron, Oded Berman, and Dmitry Krass, who were abundantly helpful and offered invaluable assistance, support and guidance. Without their knowledge and assistance this thesis would not have been finished. Special thanks also to Professor Alan Scheller-Wolf for his precious comments and encouragement as my coauthor. I am very thankful to my internal examiner, Professor Joseph Milner, for serving on my final examination committee. I also want to extend my gratitude to my external examiner Professor Costis Maglaras for his encouraging comments.

I would like to show my greatest appreciation to my friends and colleagues in the Operations Management group at the Rotman School of Management. I have greatly benefited from our pleasant and informative discussions.

I wish to express my love and gratitude to my beloved families for their understanding and endless love, through the duration of my studies.

Contents

Chapter 1

Introduction

In this thesis, we study different scheduling policies in service networks.

In Chapter 2, *Using Strategic Idleness to Improve Customer Service Experience in Service Networks*, we focus on the probability of long waits as a service level measure. The mostly common measure of service quality is the overall expected waiting time for service. However, in service networks the perception of waiting may also depend on how it is distributed among different stations. Therefore, to better capture customers' perception of service quality managers should also consider the probability of long waiting times in each station. We simultaneously consider two objectives in a service network, the expected sojourn time and the probability of long waits at any one station.

In a single-station queue it is known that the policy that minimizes the expected sojourn time and the probability of long waits is non-idling. However, in a queueing network with several stations, a policy that allows some idling may reduce the probability of long waits before any specific station. We present a family of Threshold Based Policies (TBP) that Strategically Idle (SI) some stations. We demonstrate the advantage of SI by applying TBP in a network with two single-server queues in tandem. We develop efficient algorithms to calculate the distribution of waiting time for each station and the system sojourn time under the TBP. These algorithms use an elegant analysis of the waiting time faced by specific customers. Using these results we present trade-off curves between the probability of long waits and the expected sojourn time. For the asymptotic case when $\mu_1 = \infty$, we derive closed form expressions for the performance measures.

We compare the performance of TBP policies to this of Kanban polices that are well studied idling policies in the manufacturing settings. We show that TBP policies often achieve better perform on both service level measures in the queuing networks considered.

In Chapter 3, *Dynamic Scheduling and Strategic Idling in an Open-shop Queueing Network: Case Study and Analysis*, we present a case study for improving the operations of XYZ Inc. (the real name of the company is removed for confidentially reasons). Their flagship service is the Comprehensive Health Assessment (CHA), which is composed of 10-20 different medical tests that provide customers with a complete evaluation of their current state of health and allows them to actively manage their health care. Because XYZ's customers may visit most of the stations in different orders, XYZ actually operates an open-shop queueing network.

Currently XYZ measures Customers' waiting time using three complementary Service Level (SL) measures: (i) system time—the average total time since customers register with XYZ at the beginning of the day until they finish their last exam and are free to leave, (ii) probability of system time longer than four hours, and (iii) number of red faces —the number of times (tests in a particular station) that customers have to wait more than 20 minutes. XYZ's stated SL objective targets are: mean system time of less than four hours, probability of system time longer than four hours of less than 50%, and at most five red faces per day.

The focus on the number of red faces implies that what affects service quality is indeed not just the overall waiting time, but also how this time is distributed between stations. This gives justification of considering the probability of long waits (which is a similar measure to the number of red faces) as a service level measure in Chapter 2.

In this paper, we propose an effective implementation of Dynamic Scheduling Policies (DSPs) and a generalized TBP to improve the SL in an open-shop queueing network. Using two months of data from XYZ we programmed and calibrated a detailed simulation model of XYZ's operation. Using this simulation model we demonstrate that an open-shop queueing network can be managed in a systematic fashion to deliver improved service level by jointly using DSPs and SI.

In Chapter 4, *$M/M/c$ Queue with Two Priority Classes*, we study the waiting time distribution of two different priority classes in a $M/M/c$ queue. We assume that Class-1 customers have preemptive priority over Class-2 customers. In many industries there is a growing usage

of prioritization: Companies often prioritize certain groups of customers in order to improve market segmentation, service levels, and profitability. For example, banks prioritize business customers over individual customers; websites prioritize paid users over free users; car rental companies prioritize customers with reservations over "walk-in" customers.

While the distribution of waiting times is well known for quite general single-server queues with priority such as the $M/G/1$ (see, e.g., Takagi 1991), finding this distribution is much more complicated in the multi-server setting. To the best of our knowledge, no exact solutions for the waiting time distribution in a multi-server queueing system serving multiple priority classes with *different* service rates has appeared in the literature.

The main difficulty in analyzing the $M/M/c$ queue with two priority classes is the need to track the number of jobs from each class. Thus, the state of the system is expressed in a 2D-infinite continuous-time MC, which is very difficult to analyze. We apply an innovative method to simplify the 2D-infinite continuous-time MC to a 1D-infinite discrete-time MC that is much easier to analyze. We then analyze the 1D-infinite discrete-time MC by observing the system state embedded at Class-2 departures, expressing these using a similar process to the one used in analyzing the $M/G/1$ queue. The complexity of deriving the GF increases with the number of servers, c; thus, we also provide a numerical algorithm to derive this GF for $c > 2$.

Chapter 2

Using Strategic Idleness to Improve Customer Service Experience in Service Networks

2.1 Introduction

Multi-stage service networks, where customers must visit several stations during a single service encounter, abound in modern economy. Examples range from call centers, where a typical service path may include an automated response system, followed by a generalist call-taker, and eventually (and if required) a specialist, to hospital emergency rooms, where the initial triage stage may be followed by any number of medical tests and procedures.

While there are many determinants of service quality, the link between customer waiting times and the perceived service quality is well-recognized (Friedman and Friedman, 1997; Taylor, 1994). Waiting times have long been the focus of much of the queueing literature. The most common measure of waiting time is the overall expected waiting time for service (see e.g., the survey by Gans, Koole, and Mandelbaum, 2003). A related measure is the probability that the total waiting time exceeds a certain pre-defined threshold. These measures take a macro view of the network, treating it as a one-stage system.

However, considering only such macro-level measures might not be sufficient to measure

service quality and may even be misleading. There is a strong body of evidence showing that it is also important to consider what happens within the network. A poor level of service received at a particular station may not be compensated by an exceptional service at another station, even if the overall measure appears to be acceptable. The adverse impact of long waiting time at a particular station is further supported by marketing literature, e.g., Soman and Shi (2003), and by the psychology of queueing literature, e.g., Larson (1987). Baron, Berman, and Krass (2008), Baron and Milner (2009), de-Vericourt and Jennings (2011) and references therein also focused on the probability of long waiting time as a service level measure.

Several other papers looked beyond the traditional mean waiting time measures. de-Véricourt and Zhou (2005) analyzed a call-routing problem while considering both the call resolution probability and the average service time in the overall service level measure. Mehrotra et al. (2012) considered a similar problem with heterogeneous servers. Saghafian, Hopp, and Van Oyen (2012) analyzed the service policy in Emergency Departments while considering the weighted average of the expected length of stay and the expected time to first treatment.

We recently encountered an explicit example of focusing on the probability of overly long waits at any single station at a company we call XYZ (name changed to protect confidentiality), one of the leaders in preventive healthcare services in North America. The company's primary clientele are executives and busy professionals, so it's primary focus is on providing excellent customer service experience. XYZ operates a service network with 15-20 stations. In addition to closely tracking macro-level measures, the company also records all instances where a customer waits longer than 20 minutes at a station. Any such incident results in an "red face" flashing on the manager's screen, who takes immediate steps to expedite the customer. All "red face" incidents are regarded as service failures, irrespective of whether customer's overall waiting time in the system was acceptable or not. We note that this example is not unique, e.g., the proportion of customers waiting longer than a specified time at a station is a common key performance indicator in call centers. The focus on long waits implies that service quality is affected not only by the overall waiting time, but also by the distribution of waiting among stations.

The focus of this paper is to simultaneously consider two objectives in a service network, one based on some macro-level measure and one based on the *probability of excessive wait* at any

one station. The difference in managing these two objectives can be rather dramatic. Indeed, the macro-level service measures are typically minimized by using work-conserving policies, where system resources are not idled as long as there is work in the system. Such policies are optimal with respect to minimizing overall service times and are the focus of most studies of queueing networks (see, e.g., Chen and Yao 2001 and reference therein). However, using a work-conserving policy is not necessarily a good idea when it comes to the second objective. Consider a situation where one station in the network accumulates a long queue, while the waiting times are low at the upstream stations. In such a case, continuing to operate upstream stations at the normal rate may increase the probability of excessive waits at downstream stations. A better idea may be to temporarily reduce the service rate or idle the upstream stations, allowing the downstream queue to dissipate. By intentionally idling some resources we are effectively re-distributing the waiting times more evenly within the network. As long as such redistribution does not significantly increase the overall system times (i.e. the first objective), it may well improve the overall customer service experience.

Our objective is to propose and analyze a class of scheduling policies that intentionally idle some resources in order to reduce the probability of excessive waits at any one station. We refer to such intentional idling of resources as *strategic idleness* (SI). Note that the classical way of reducing waiting time and probabilities of long waits is to add resource capacity to the system (e.g., adding a doctor in the healthcare setting), which is often quite expensive. On the other hand, changing the scheduling rules to intentionally idle some resources can often be done at a negligible cost. Thus, a switch to an SI policy may be very cost-effective of improving customer service experience. Indeed, we establish that in contrast to the single station queue, where a non-idling scheduling policy minimizes both the sojourn time and the probability of long waits, for a multi-stage queueing network policies with SI may significantly reduce the probability of long waits while only slightly increasing the overall time in the system. To the best of our knowledge, ours is the first paper to systematically study SI as a mechanism for reducing the probability of excessive waits and improving the customer service experience.

In service networks, long waits can be measured in a variety of ways. For example, consider a two-station tandem queue with Station 1 as the upstream machine and Station 2 as the downstream one. The specific measure we consider is $PW(t) = \frac{1}{2}\sum_{i=1}^{2} P\{W_i > t\}$, where W_i

is the steady state customers' waiting time for station i, and t is the time threshold designating an "excessive wait". We interpret $PW(t)$ as the frequency with which customers experience excessive waits. We note that in place of $PW(t)$ one can use other related measures, e.g., $1 - P\{W_1 < t, W_2 < t\}$, i.e., the probability that a customer experiences at least one excessive wait.

There are many possible policy classes that involve SI. Our primary focus is on a specific family of *Threshold Based Policies (TBP)*. The idea behind the TBP is simple, it compares the difference between queue lengths at different stations and idles some upstream stations if this difference is larger than a predetermined threshold. For example, consider the two-station tandem queue described above: let q_1, q_2 be the lengths of queues in front of the respective stations. A TBP, defined by the value of the threshold TH, idles Station 1 whenever the difference $q_2 - q_1 \geq TH$ (we only consider $TH \geq 0$ as using $TH < 0$ is clearly counterproductive, e.g., with $TH = -1$ when $q_1 = 1, q_2 = 0$, Station 1 would be idled).

We note that, assuming Poisson arrivals to Station 1 and exponentially distributed and independent service times at both stations, the performance of the non-idling (NI) policy is easy to analyze (see e.g. Ross 2000, Chapter 8). However, such an analysis for the system operating under the TBP is quite challenging for several reasons. First, the process is not reversible, so arrivals to Station 2 do not follow a Poisson process. Second, as explained in Section 2.3, customer's waiting time for Station 1 depends on future arrivals, so Little's Distributional Law (see e.g., Bertsimas and Nakazato (1995) and Bertsimas and Mourtzinou (1996)) does not hold.

We develop efficient algorithms to calculate the distribution of waiting time for each station and the system sojourn time under the TBP. These algorithms use a novel analysis of the waiting time faced by specific customers. Using these results we present trade-off curves between the probability of long waits and the expected sojourn time. (Note that the distribution of the system sojourn time can provide other measures than the mean, but the trade-offs between $PW(t)$ and these measures are similar to the trade-off between $PW(t)$ and the mean sojourn time.) For the asymptotic case when $\mu_1 = \infty$, we derive closed form expressions for the performance measures. We derive interesting insights that also hold in the case of finite processing capacity for both stations.

Our results show that TBP can significantly reduce the probability of long waits (as ex-

pressed by $PW(t)$ or similar measures) versus the NI policy as long as the waits of length t are sufficiently rare in the system. If, on the other hand, the frequency of such "excessive" waits is high under the NI policy (indicating that they are not, in fact, excessive), then the TBP is unlikely to provide an improvement - the only way to decrease such waits is by adding capacity.

We also consider the class of TBPs in a tandem queue network with three stations. By developing a simulation model, we show that a TBP can reduce the probability of long waits while only slightly increasing sojourn times. A comparison with Kanban policies indicates that the TBP perform significantly better in this case.

We note that service systems, such as XYZ, do not always reach steady state before the end of a business day. Moreover, such systems often operate a non-serial queueing network. However, the results for the serial system under the steady state assumption still provide valuable insights for such systems. Specifically, polices with SI such as the TBP can improve customers' perception of the service level with little cost. In Baron et al. (2013), we tested a generalized TBP with a simulation model of the open-shop operation of XYZ; we indeed established that TBP can be effective in improving customers' perception of the service level.

The outline of the paper is as follows. In the next section, we provide a brief discussion of other policies with idling. After introducing the TBP for the 2-station network in Section 2.3, we consider the asymptotic $\mu_1 = \infty$ case in Section 2.4. In Section 2.5, we analyze the case of finite processing rate for both stations. In Section 2.7, we discuss generalization of the TBPs, to n-station serial queues and list several open questions. All proofs are in Appendix 2.9.1.

2.2 Literature Review - Other Policies with Idling

Note that the main idea behind TBP - idling an upstream station when a downstream station is facing a large workload - can be achieved by other policy classes. We next briefly review classes of policies that are discussed in the literature of manufacturing systems.

Masin, Herer, and Dar-el (2010) developed a unified model that encompasses and compares a wide range of production control policies. We follow their exposition focusing on a serial manufacturing system with M stations, and each station i has an input pile, IP_i, and an output pile, OP_i, for $i = 1, ..., M$. OP_0 represents an ample pile of raw materials, i.e., $OP_0 = \infty$. Each

part waits in IP_i before being processed at station i and then transferred to OP_i; and stays in OP_i until it can be transferred to IP_{i+1}.

There are four well known static control policies (i.e., controls that are independent of the system state) are: *Fixed Buffer* policy (see, e.g., Conway et al. 1988) places a finite buffer FB_{i+1} between stations i and $i+1$, i.e., $IP_1 < FB_1$ and $OP_i + IP_{i+1} \leq FB_{i+1}$ for $i = 1, ..., M-1$; *Kanban* policy, implemented by Toyota (Sugimori et al. 1977), places an upper bound KB_i on the total number of parts associated with station i, i.e., $IP_i + OP_i \leq KB_i$ for $i = 1, ..., M$; Constant work in process (*CONWIP*) policy, first presented by Spearman et al. (1990), places an upper bound CW on the total number of parts in the system, i.e., $\sum_{j=1}^{M} (IP_j + OP_j) \leq CW$ (For a recursive calculation of several performance measure in a resulting closed queueing network see Sloberg (1977)); *Base-stock* policy (see, e..g., van Ryzin et al. 1993), places an upper bound BS_i on the total number of parts at the downstream of station i, i.e., $\sum_{j=i}^{M} (IP_j + OP_j) \leq BS_i$ for $i = 1, ..., M$.

More sophisticated dynamic control policies where controls depend on the state of the system were also studied. Weber and Stidham (1987) considered a general model for control of service rates ($\mu_i \in [0, \bar{\mu}_i]$) in a serial or closed queueing network, where control policies depend on the entire state vector $q = (q_1, q_2, ..., q_M)$ where $q_i = OP_{i-1} + IP_i$. They considered the sum of total inventory holding cost and stations operating cost as the objective function. They provided necessary conditions, called the "monotonicity result", for any control policy to be optimal: 1) the optimal service rate at station i does not decrease as a customer finishes service at another station; 2) the optimal service rate at station i does not increase as a customer finishes service at station i. They apply their monotonicity result to models where stations can only be turned on or off ($\mu_i = 0$ or $\bar{\mu}_i$) and show that it is optimal to turn an off-station on as the numbers of customers at its downstream stations decrease, or as the numbers of customers at upstream stations increases. Note that the four control policies discussed above and TBP all satisfy this monotonicity result. Veatch and Wein (1994) considered the optimal control of a two-station tandem production/inventory system with a similar objective function. They compared these four policies, gave conditions under which certain simple controls are optimal, and computed the dynamic optimal controls using dynamic programming.

There are several conceptual differences between the control policies discussed above, tai-

lored to manufacturing systems, and the TBP, tailored to service systems. First, the main motivation behind developing policies in manufacturing setting is the control of expected inventory costs. This motivation is different for service systems focusing on the effect of the distribution of waiting time on customers' experience. As we demonstrate below, this different motivations also leads to a different analysis. In fact, to the best of our knowledge, no analysis of the distribution of waiting times under the policies mentioned above is available; such an analysis appears to be subject to many of the challenges as in the analysis of the TBP. Second, another important modeling difference is that the control for manufacturing systems is often modeled as a make-to-stock system, whereas the control for service systems must be modeled as a make-to-order system. Third, from a modeling perspective, the supply and demand models are also different in a service system: the service at a first station is initiated by an exogenous arrival process and customers leave the system as they complete service at the last station, whereas in manufacturing the exogenous demand arrives to the last station. A final difference is with respect to admission control. In contrast to our model, where all customers are accepted, models for manufacturing system often operate with admission control where not all arriving orders are fulfilled. (Note that IP_1 is bounded in the four policies above, so *not all* arriving customers are admitted. Still, if all customers need to be admitted, IP_1 can be removed from all constraints. For example, a CONWIP policy could place an upper bound CW on the total number of parts without considering IP_1, i.e., $OP_1 + \sum_{j=2}^{M} (IP_j + OP_j) \leq CW$.)

Despite these differences, the control policies developed for manufacturing systems can be applied in service systems (sometimes with a few modifications). When applied in a two-station tandem queue service system without admission control, the Fixed Buffer, Kanban, CONWIP and Base-stock policies can all be shown to be equivalent. To illustrate the equivalence of Kanban policy and Fixed Buffer policy note that a Kanban policy with KB_1 and KB_2 is equivalent to a Fixed Buffer policy with buffer size $FB_2 = KB_1 + KB_2$ between the two stations; and a Fixed Buffer policy with buffer size FB_2 is equivalent to a Kanban policy with $KB_1 = 1$ and $KB_2 = FB_2 - 1$. Thus, in the 2-station tandem queue service system we consider in the paper, we focus on a Kanban policy that idles Station 1 whenever $q_2 \geq BS$, where BS is the size of the buffer between the two stations.

In this paper, we compare our TBP with the Kanban policy. Note that in the 2-station

case, our TBP is a more sophisticated dynamic control policy, where the upper bound of q_2 is a linear function of q_1, i.e., Station 1 is idled whenever $q_2 \geq q_1 + TH$; and a Kanban policy idles Station 1 based only on $q_2 \geq BS$ irrespective of the value of q_1, and thus - intuitively - it provides less flexible control than a TBP. This intuition appears to be supported by our results. For the asymptotic case when Station 1 has infinite processing capacity we derive closed form expressions for the $PW(t)$ measure under a Kanban policy, allowing us to make analytical comparisons to a TBP. For the finite capacity case we use Monte Carlo simulation to compare TBP and Kanban policies. Our results indicate that, similar to TBP, Kanban policy allows for the trade-off between the $PW(t)$ measure and expected service times. However, this policy appears to be less efficient than the TBP.

In closing this section we note that (i) the idea of intentionally idling a capacitated resource has also been considered by Afèche (2013). In the revenue management context, he showed how such delays can allow a seller to differentiate between customer types and thus improve the overall profit. His motivation and analysis are much different than ours. (ii) Recent polices for control of manufacturing systems often considered prioritization among several customer classes, but are focused on a single stage system. Ha (1997a, b) was the first to discuss inventory rationing problems in a centralized make-to-stock system. He focused on base stock level production control. (iii) In the revenue management context, Caldentey and Wein (2006) developed a diffusion approximation for profit maximization with two classes of customers. They show that a dynamic control policy based upon the inventory or backlog level is effective.

Finally, we are aware that there are other policies that consider the entire system state. This paper serves as a stepping stone motivating the analysis of such policies in service systems.

2.3 Two Queues in Tandem - Preliminary Analysis

Consider the two-station tandem queueing network with two sequential single server stations and infinite buffer space discussed before. We define a simple TBP for this network as follows: upon completing service, Station 1 is idled and will not admit the next customer to service if

$$\delta(q_1, q_2) = q_2 - q_1 \geq TH.$$

Station 1 will resume work once $\delta(q_1, q_2) < TH$. When no ambiguity arises, we will use δ instead of $\delta(q_1, q_2)$. We denote $TBP(TH)$ as the TBP with threshold TH. We say that a customer is *stopped* (at Station 1) if this customer is waiting at Station 1 while this station is *idled*.

Three events can occur in this tandem queueing network:

1. *Arrival* - arrival to the network decreases δ by 1. Arrivals occur with rate λ at any state.

2. *Completion 1* - service completion at Station 1 increases δ by 2. This happens with rate μ_1, if $q_1 \geq 1$ and $\delta < TH$ (when Station 1 is not idled).

3. *Completion 2* - service completion at Station 2 decreases δ by 1. This event has rate μ_2, if $q_2 \geq 1$.

Note that since δ decreases when Station 2 completes service or when a new customer arrives to Station 1, either of these two events may cause Station 1 to resume work.

From these three events, we conclude that there are two situations when Station 1 is idled: $\delta = TH$ or $\delta = TH + 1$. When $\delta = TH + 1$, Station 1 is idled, so only Arrival or Completion 2 can happen in the network. After a time period, which is distributed $\sim \exp(\lambda + \mu_2)$, one of these events happen, reducing δ to TH. Note that Station 1 remains idled. This sequence repeats and after another time period $\sim \exp(\lambda + \mu_2)$, δ is reduced to $TH - 1$, at which point Station 1 resumes work, and its idle period ends. We define *stoppage* as the time period from the moment when the value of δ changes and Station 1 becomes idled until the moment when either Arrival or Completion 2 happens. With this definition, when $\delta = TH + 1$, customers in Station 1 experience two stoppages before Station 1 resumes work; when $\delta = TH$, they only experience one stoppage.

Let $Q_i(t)$, $i = 1, 2$ be the random variable denoting the total number of customers at Station i (in queue and in service) at time t. Given TH, the process $(Q_1(t), Q_2(t))$ is a continuous time Markov Chain (MC). Let π_{q_1,q_2} denote the steady state probability of $MC(Q_1, Q_2)$. Let S be the sojourn time for any customer, i.e., $S = \textit{total waiting time} + \textit{total service time}$.

To investigate the trade-off between $PW(t)$ and $E[S]$ under the TBP, we first characterize the distribution of three steady state service measures: the waiting time at Station 1, W_1; the

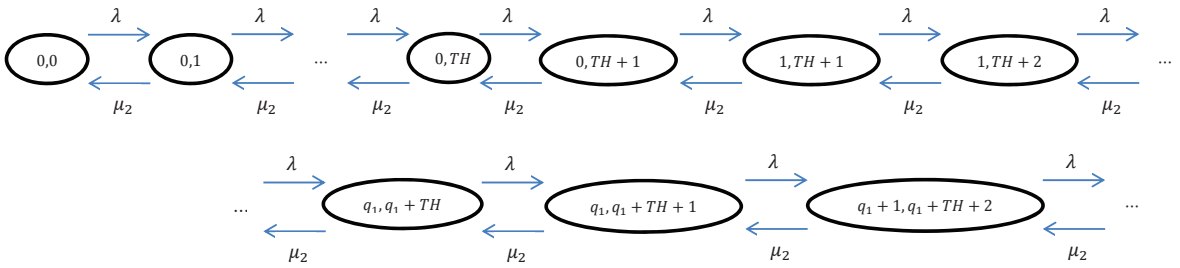

Figure 2.1: MC when $\mu_1 = \infty$ and $TH > 0$.

waiting time at Station 2, W_2, and the sojourn time, S. We can calculate the distributions of these three measures by conditioning on the state (q_1, q_2) seen by a random arrival. Let X^{q_1,q_2} be any one of these three measures experienced by a *tagged customer* (TC) who arrives in state (q_1, q_2). Then, the steady state distribution of X can be calculated as

$$\begin{aligned} P\{X > t\} &= \sum_{q_1,q_2} P\{X^{q_1,q_2} > t \mid \text{TC sees } (q_1,q_2) \text{ at arrival}\} P\{\text{TC sees } (q_1,q_2) \text{ at arrival}\} \\ &= \sum_{q_1,q_2} P\{X^{q_1,q_2} > t \mid \text{TC sees } (q_1,q_2) \text{ at arrival}\} \pi_{q_1,q_2}, \end{aligned} \tag{2.1}$$

where the second equality follows by PASTA.

Similar to (2.1), the Laplace Transform (LT) of X can be written as

$$L_X(h) = \sum_{q_1,q_2} L_{X^{q_1,q_2}}(h \mid \text{TC sees } (q_1,q_2) \text{ at arrival}) \pi_{q_1,q_2}. \tag{2.2}$$

2.4 Asymptotic Case: Station 1 Has an Infinite Service Capacity

We next calculate the steady state performance measures under the TBP and compare them with the measures for the non-idling network and the Kanban policy when Station 1 has infinite capacity. For convenience, we denote quantities related to this asymptotic case with a ˆ, e.g., $\hat{W}_i$ is the waiting time at station i. A full list of notation can be found on Table 2.9.5 in Appendix 2.9.5.

The MC for the $\mu_1 = \infty$ case is depicted on Figure 2.1. As described in Section 2.3, three

events occur in this MC: Arrival, Completion 1, and Completion 2. However, since Completion 1 happens instantaneously, only two events are shown on the figure: Arrival (at rate λ) and Completion 2 (at rate μ_2). Consider the state $(0, TH)$, where Station 1 is idled under the TBP. An Arrival momentarily bring the MC to state $(1, TH)$, where $\delta = TH - 1$ and thus Station 1 resumes work, instantaneously bringing the MC to state $(0, TH + 1)$, and idling Station 1 again. At the next Arrival the MC transitions to state $(1, TH + 1)$ where $\delta = TH$ and thus the newly arrived customer is stopped. This stoppage lasts until either a new Arrival, which allows the system to process the first customer from Station 1 and sends the system to state $(1, TH + 2)$, or Completion 2, which also releases the customer from Station 1 and sends the system to $(0, TH)$. In general, whenever $q_1 > 0$, Station 1 is idled and the system is either in state $(q_1, q_1 + TH)$ or $(q_1, q_1 + TH + 1)$.

The steady-state distribution of this simple Birth and Death MC (similar to the solution of $M/M/1$ queue), for $q_1 = 0, q_2 = 0, ..., TH + 1$ and for $q_1 > 0, q_2 = q_1 + TH, q_1 + TH + 1$, is:

$$\pi_{q_1,q_2} = \rho_2^{q_1+q_2}(1 - \rho_2). \tag{2.3}$$

Remark 1 *If we consider $q_1 + q_2$ as the total queue length, this network has the same steady state probability distribution as a $M/M/1$ queue with $\rho_2 = \frac{\lambda}{\mu_2}$. Because Station 2 works as long as there are customers in the network, the sojourn time is the same as the sojourn time in the system with $\mu_1 = \infty$ operating under a non-idling policy. Thus, in the asymptotic case the TBP does not increase the sojourn times and we can focus solely on the $PW(t)$ measure.*

Remark 2 *Suppose the system is in state $(q_1, q_1 + TH + 1)$ for $q_1 > 0$ (Station 1 is idled). The next Arrival (Completion 2) event sends the system to state $(q_1 + 1, q_1 + TH + 1)$ (state $(q_1, q_1 + TH)$), with $\delta = TH$, and Station 1 is stopped again. Thus, the next event must be another Arrival or Completion 2. Similarly, suppose the system is in state $(q_1, q_1 + TH)$ for $q_1 > 0$ (Station 1 is idled). The next event must be Arrival or Completion 2, which will trigger a Completion 1 event and send the system to $(q_1, q_1 + TH + 1)$ or $(q_1 - 1, q_1 + TH)$, respectively, with $\delta = TH + 1$ in both cases. Thus, as long as $q_1 > 0$, between any two Completion 1 events there are always two other events. This leads to the following Proposition.*

Proposition 1 *Let $\hat{M}^{q_1,q_2}$ be the number of stoppages TC sees before entering Station 2, given she arrives in state (q_1, q_2). Then either $q_2 < TH$ and $\hat{M}^{q_1,q_2} = 0$, or $q_2 \in \{q_1+TH, q_1+TH+1\}$ and $\hat{M}^{q_1,q_2} = q_1 + q_2 - TH$.*

2.4.1 Distribution of $\hat{W}_1$, Waiting Time at Station 1

In general, the TC's waiting time for Station 1 is composed of two parts: the service time of customers in front of her in Station 1 and the stoppages of Station 1. However, when $\mu_1 = \infty$, the service time of Station 1 is zero, and thus $\hat{W}_1$ is only caused by stoppage.

Let $\hat{W}_1^{q_1,q_2}$ denote the TC's waiting time at Station 1, given that she arrives at state (q_1, q_2). From Proposition 1, if $q_1 + q_2 \leq TH$, TC sees no stoppage and $\hat{W}_1^{q_1,q_2} = 0$; similarly, if $q_1 + q_2 > TH$, then $\hat{M}^{q_1,q_2} = q_1 + q_2 - TH$, so that $\hat{W}_1^{q_1,q_2}$ is distributed as $Erlang(\lambda + \mu_2, q_1 + q_2 - TH)$. Thus, using (2.2) and (2.3) the LT of $\hat{W}_1$ is

$$\begin{aligned} L_{\hat{W}_1}(h) &= \sum_{i=0}^{TH} \rho_2^i(1-\rho_2) + \sum_{i=TH+1}^{\infty} \rho_2^i(1-\rho_2)\left(\frac{\lambda+\mu_2}{\lambda+\mu_2+h}\right)^{i-TH} \\ &= \left(1-\rho_2^{TH+1}\right) + \rho_2^{TH+1}\frac{(\mu_2-\lambda\rho_2)}{(\mu_2-\lambda\rho_2)+h}. \end{aligned} \tag{2.4}$$

From the transform of $\hat{W}_1$ we conclude that there is no waiting in Station 1 with probability $\left(1-\rho_2^{TH+1}\right)$, and the waiting is distributed as an $\exp(\mu_2 - \lambda\rho_2)$ R.V. with probability ρ_2^{TH+1}. Hence,

$$P\left\{\hat{W}_1 > t\right\} = \rho_2^{TH+1} e^{-(\mu_2-\lambda\rho_2)t}. \tag{2.5}$$

Note that given waiting (i.e., with probability ρ_2^{TH+1}) $\hat{W}_1$ is distributed as the waiting time given waiting in an $M/M/1$ queue with arrival rate $\lambda\rho_2$ and service rate μ_2.

As intuition suggests $P\left\{\hat{W}_1 > t\right\}$ is a decreasing function of TH. When TH decreases, customers see more stoppages, and thus wait more in Station 1. When TH increases, the TBP's effect on the network is reduced and customers' wait in Station 1 is also reduced. The extreme case when $TH = \infty$ results in a non-idling network, so customers do not wait for Station 1.

2.4.2 Distribution of the Waiting Time $\hat{W}_2$ and Service Measure $P\hat{W}(t)$.

We next derive $\hat{W}_2^{q_1,q_2}$, the TC's waiting time at Station 2 given that she arrives at state (q_1, q_2), and then use (2.2) to calculate the LT of $\hat{W}_2$. Let K be the R.V. denoting (we omit the dependency in q_1, q_2) the number of customers in Station 2 when the TC enters this station; thus $\hat{W}_2^{q_1,q_2}$ is distributed as $Erlang(\mu_2, K)$.

From Proposition 1, if $q_1+q_2 \leq TH$, then $q_1 = 0$ and the TC gets into Station 2 immediately implying that $K = q_1 + q_2 = q_2$. Thus, for $q_1 + q_2 \leq TH$ the distribution of $\hat{W}_2^{q_1,q_2}$ is $Erlang(\mu_2, q_1 + q_2)$ with the LT given by

$$L_{\hat{W}_2^{q_1,q_2}}(h) = \left(\frac{\mu_2}{\mu_2 + h}\right)^{q_1+q_2}. \tag{2.6}$$

Now suppose the TC arrives at state (q_1, q_2) with $q_1 + q_2 > TH$, implying that the number of stoppages $\hat{M}^{q_1,q_2} = q_1 + q_2 - TH$. In this case, $\hat{M}^{q_1,q_2}$ Arrival or Completion 2 events are required to end these stoppages, and $q_1 + q_2 - K$ of these are Completion 2 events, so $K \in [TH, q_1 + q_2]$. Since the probability that the next event is an Arrival (Completions 2) is $\frac{\lambda}{\lambda+\mu_2}$ ($\frac{\mu_2}{\lambda+\mu_2}$), it follows that $q_1 + q_2 - K$ has the binomial distribution:

$$P\{q_1 + q_2 - K = n\} = \binom{q_1 + q_2 - TH}{n} \left(\frac{\lambda}{\lambda + \mu_2}\right)^{q_1+q_2-TH-n} \left(\frac{\mu_2}{\lambda + \mu_2}\right)^n, \; n = 0, ..., q_1+q_2-TH.$$

Thus

$$P\{K = k\} = \binom{q_1 + q_2 - TH}{q_1 + q_2 - k} \left(\frac{\lambda}{\lambda + \mu_2}\right)^{k-TH} \left(\frac{\mu_2}{\lambda + \mu_2}\right)^{q_1+q_2-k}, \; k = TH, ..., q_1 + q_2.$$

Therefore, for $q_1 + q_2 > TH$ the LT of $\hat{W}_2^{q_1,q_2}$ is

$$\begin{aligned} L_{\hat{W}_2^{q_1,q_2}}(h) &= \sum_{k=TH}^{q_1+q_2} \binom{q_1 + q_2 - TH}{q_1 + q_2 - k} \left(\frac{\lambda}{\lambda + \mu_2}\right)^{k-TH} \left(\frac{\mu_2}{\lambda + \mu_2}\right)^{q_1+q_2-k} \left(\frac{\mu_2}{\mu_2 + h}\right)^k \\ &= \left(\frac{\mu_2}{\mu_2 + h}\right)^{TH} \left(\frac{\mu_2}{\lambda + \mu_2} + \frac{\lambda}{\lambda + \mu_2}\frac{\mu_2}{\mu_2 + h}\right)^{q_1+q_2-TH} \\ &= \left(\frac{\mu_2}{\mu_2 + h}\right)^{q_1+q_2} \left(\frac{\lambda + \mu_2 + h}{\lambda + \mu_2}\right)^{q_1+q_2-TH}. \end{aligned} \tag{2.7}$$

The second equality follows Binomial Formula. The third equality follows because for $q_1+q_2 > TH$,

$$\left(\frac{\mu_2}{\mu_2+h}\right)^{TH}\left(\frac{\lambda\mu_2}{(\lambda+\mu_2)(\mu_2+h)}+\frac{\mu_2}{\lambda+\mu_2}\right)^{q_1+q_2-TH}\left(\frac{\lambda+\mu_2}{\lambda+\mu_2+h}\right)^{q_1+q_2-TH}=\left(\frac{\mu_2}{\mu_2+h}\right)^{q_1+q_2}. \tag{2.8}$$

We can now write the LT of $\hat{W}_2$ using (2.2), (2.3), (2.6) and (2.7):

$$L_{\hat{W}_2}(h)=\sum_{i=0}^{TH-1}\rho_2^i(1-\rho_2)\left(\frac{\mu_2}{\mu_2+h}\right)^i+\sum_{i=TH}^{\infty}\rho_2^{2i-TH}(1-\rho_2^2)\left(\frac{\mu_2}{\mu_2+h}\right)^i. \tag{2.9}$$

From the LT of $\hat{W}_2$ we know that $\hat{W}_2$ is distributed as a $Erlang(\mu_2, q_1+q_2)$ R.V. with probability $\rho_2^{q_1+q_2}(1-\rho_2)$, for $0 \le q_1+q_2 < TH$, (i.e., when the TC experiences no stoppages); and as the sum of an $Erlang(\mu_2, TH-1)$ R.V. and an $\exp(\mu_2-\lambda\rho_2)$ R.V. with probability ρ_2^{TH}. Using (2.9), we can derive the Tail Distribution of $\hat{W}_2$ under the TBP with threshold TH:

$$P\left\{\hat{W}_2 > t\right\} \tag{2.10}$$

$$= \begin{cases} \rho_2^{2-TH}e^{-(\mu_2-\lambda\rho_2)t} \ if\ TH=0,1 \\ \rho_2^{2-TH}e^{-(\mu_2-\lambda\rho_2)t}+\rho_2 e^{-\mu_2 t}\sum_{k=0}^{TH-2}\frac{(\mu_2 t)^k}{k!}\rho_2^k-\rho_2^2 e^{-\mu_2 t}\rho_2^{-TH}\sum_{k=0}^{TH-2}\frac{(\mu_2 t)^k}{k!}\rho_2^{2k} \ if\ TH\ge 2 \end{cases} \tag{2.11}$$

Using (2.5) and (2.11), the distribution of our main service level measure under the TBP with threshold TH is

$$\begin{aligned} PW^{TBP(TH)}(t) &= \frac{1}{2}\left(P\left\{\hat{W}_1 > t\right\}+P\left\{\hat{W}_2 > t\right\}\right) \\ &= \begin{cases} \frac{1}{2}\rho_2^2 e^{-(\mu_2-\lambda\rho_2)t}+\frac{1}{2}\rho_2 e^{-(\mu_2-\lambda\rho_2)t} \ if\ TH=0,1 \\ \frac{1}{2}\rho_2^{TH+1}e^{-(\mu_2-\lambda\rho_2)t}+\frac{1}{2}\rho_2 e^{-\mu_2 t}\sum_{k=0}^{TH-2}\frac{(\rho_2\mu_2 t)^k}{k!} \\ +\frac{1}{2}\rho_2^{2-TH}e^{-(\mu_2-\lambda\rho_2)t}-\frac{1}{2}\rho_2^{2-TH}e^{-\mu_2 t}\sum_{k=0}^{TH-2}\frac{(\mu_2 t)^k}{k!}\rho_2^{2k} \ if\ TH\ge 2 \end{cases} \end{aligned} \tag{2.12}$$

We observe that under the non-idling policy all waiting happens at Station 2 and thus

$$PW^{NI}(t)=\frac{1}{2}\rho_2 e^{-(\mu_2-\lambda)t},\ t>0. \tag{2.13}$$

Here, ρ_2 represents the probability of waiting and $exp(-(\mu_2 - \lambda)t)$ is the conditional probability of a waiting more than t given an $M/M/1$ queue with parameters (λ, μ_2). In the expression for $PW^{TBP}(t)$ when $TH = 0, 1$ we see the same structure as in (2.13). The first term, essentially has the probability of waiting reduced to ρ_2^2 from ρ_2 and the arrival rate reduced to $\rho_2\lambda$ from λ. The Second term, is just the probability of wait longer than t in an $M/M/1$ queue with arrival rate $\rho_2\lambda$. Thus, the TBP effectively operates two $M/M/1$ stations with parameters $(\rho_2\lambda, \mu_2)$, where the probability of wait at one of these stations is further reduced by ρ_2. The slower arrival rate (and the additional reduction in probability of waiting) brings the probability of wait longer than t at Station 2 to below the level experienced at this Station under the non-idling policy. However, customer now has two chances to experience a long wait - once at each station.

2.4.3 Insight 1: Comparing TBP and Non-Idling Policy for the Asymptotic Case

Based on Remark 1 above, it suffices to compare $PW^{TBP}(t)$ with $PW^{NI}(t)$ since expected service times is the same. From our earlier discussion, it is obvious that the number of stoppages is increased when TH is reduced. Thus, setting $TH = 0$ corresponds to the most aggressive redistribution of the waiting time from Station 2 to Station 1 achievable by a TBP (from (2.12)). On the other hand, $PW^{TBP(\infty)}(t) = PW^{NI}(t)$ since when $TH = \infty$, Station 1 is never intentionally idled.

For any "excessive wait" value $t > 0$ let $TH^*(t) = \arg\min_{TH} PW^{TBP(TH)}(t)$ be the threshold value that minimizes $PW(t)$. This value is characterized in the following result.

Proposition 2 *For any t, the threshold $TH^*(t) \in \{0, \infty\}$. Specifically, let $t^* = \frac{\ln(1+\rho_2)}{\lambda(1-\rho_2)}$ (note that $PW^{TBP(0)}(t^*) = \frac{\rho_2}{2}(\rho_2 + 1)^{-\frac{1}{\rho_2}}$). If $t \leq t^*$, then $TH^*(t) = \infty$, and if $t > t^*$, then $TH^*(t) = 0$.*

This Proposition indicates that the optimal TBP is to idle Station 1 as much as possible when t is sufficiently large (i.e., use $TH^* = 0$ when $t > t^*$), or to not idle it at all when t is small (i.e., $t \leq t^*$). The intuitive explanation behind this is that reducing the queue sizes at Station 2 via the TBP reduces $P(\hat{W}_2 > t)$ but introduces $P(\hat{W}_1 > t) > 0$ (which is 0 under the

NI policy). When t is large, the reduction in $P(\hat{W}_2 > t)$ is substantial, while the increase in $P(\hat{W}_1 > t)$ is small, and thus TBP outperforms the NI policy. However, if t is small, the waits longer than t are quite common at Station 2 even if some customers are re-allocated to Station 1, while the increase in $P(\hat{W}_1 > t)$ may be substantial. Thus $TH^* = \infty$ and TBP is equivalent to the NI policy. In this case the re-allocation of waiting time will not solve the problem of excessive waits - the only solution is adding more capacity to the system.

From (2.12) and (2.13), the reduction in $PW(t)$ due to TBP for $t > t^*$ is:

$$\frac{PW^{NI}(t) - PW^{TBP(0)}(t)}{PW^{NI}(t)} = 1 - (1+\rho_2)\, e^{-\lambda(1-\rho_2)t}.$$

Thus, the relative improvement in $PW(t)$ increases with t, and approaches 100% as t increases. This shows that the TBP can dramatically reduce the incidence of excessive waits, but only if the designation of an "excessive" wait is used correctly, i.e., a wait is "excessive" if it is uncommon in the system.

The implications for the decision-maker are clear: if waits of at least t adversely affect customer service experience, and $t > t^*$, TBP can be used to improve $PW(t)$. If $t \leq t^*$, then the only way to improve $PW(t)$ is by adding capacity to the system (i.e., increasing μ_2). Most of the behaviors observed for the $\mu_1 = \infty$ case will also hold for the $\mu_1 < \infty$ case discussed in Section 2.5.

2.4.4 Insight 2: Comparing TBP and Kanban Policy for the Asymptotic Case

For the 2-station tandem queue a Kanban policy is defined by the buffer size ($BS \geq 1$) in front of Station 2: Station 1 is idled and will not admit the next customer to service whenever $q_2 \geq BS$.

Observe that in the $\mu_1 = \infty$ case, under Kanban(BS) policy Station 2 operates as long as there are customers in the system for any $BS \geq 1$. Thus the expected sojourn time for any Kanban policy is the same as for NI policy. Therefore, as in the TBP case, we focus only on $PW(t)$.

Using similar analysis as for the TBP, we have:

Proposition 3 *For* $BS \geq 1$,

$$PW^{Kanban(BS)}(t) = \begin{cases} \frac{1}{2}\rho_2 e^{-(\mu_2-\lambda)t} & \text{if } BS = 1 \\ \frac{1}{2}\rho_2^{BS} e^{-(\mu_2-\lambda)t} + \frac{1}{2}e^{-\mu_2 t}\sum_{k=0}^{BS-2}\frac{(\mu_2 t)^k}{k!}\rho_2^{k+1} & \text{if } BS \geq 2 \end{cases}. \quad (2.14)$$

Note that $PW^{Kanban(1)}(t) = PW^{NI}(t) = PW^{Kanban(\infty)}(t)$. This is because when $BS = 1$, the Kanban policy shifts all waiting time to Station 1 without changing the distribution of waiting times. This shows that by optimizing the buffer size, a Kanban policy can outperform NI with respect to the $PW(t)$ measure. The second equation holds because when $BS = \infty$, Station 1 is never idled.

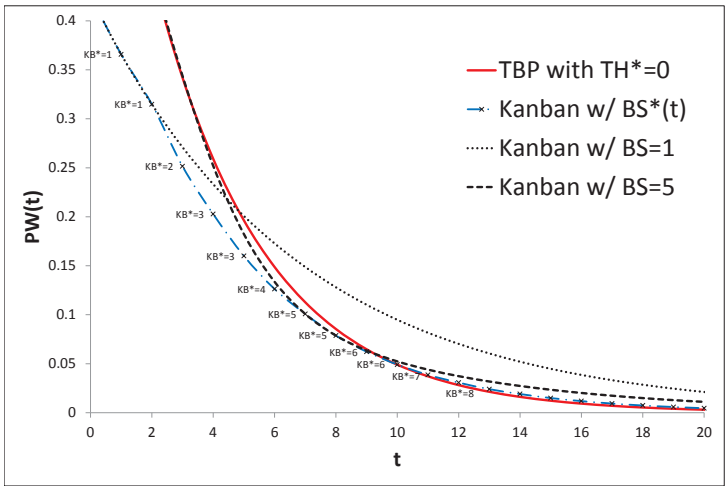

Figure 2.2: $PW(t)$ as a function of t under TBP versus Kanban policy.

In Figure 2.2 we compare $PW^{TBP(0)}(t)$ with $PW^{Kanban}(t)$ under different BS values, when $\lambda = 0.85$ and $\mu_2 = 1$. Note that $t^* = 4.82$ in this case and thus TBP(0) outperforms the NI policy for $t > 4.82$. Recalling that Kanban(1) policy is equivalent to NI, we see that this is indeed the case on Figure 1, with the relative gap growing with t. Comparing TBP(0) with Kanban(5) we see that TBP has lower $PW(t)$ for $t > 9.28$, while Kanban performs better for lower values of t.

Furthermore, using a similar analysis to the one in the proof of Proposition 2, we can obtain the buffer size $BS^*(t)$ that minimizes $PW^{Kanban(BS)}(t)$ for any t. (Specifically, the function $g_4 = \frac{(\mu_2 t)^{BS-1}}{(BS-1)!} - (1-\rho_2)e^{\lambda t}$ has one or two zero points; if g_4 has two zero points, $BS^*(t)$ is the smaller zero point of g_4, otherwise $BS^*(t) = 1$.) The resulting Kanban($BS^*(t)$) policy

is plotted on Figure 2.2 along with the associated $BS^*(t)$ values. This policy achieves lower $PW(t)$ values than the TBP(0) for $t < 9.96$ and slightly higher values for $t > 9.96$.

For the asymptotic case the Kanban policies perform competitively with TBP(0), particularly when the buffer size is optimized for a given t value. We note that the TBP is more robust - as the same optimal threshold $TH^* = 0$ value applies over a wide range of t values, while the optimal buffer size $BS^*(t)$ is sensitive to t. More importantly, the performance of Kanban policies in the asymptotic case are somewhat misleading, we will see in the following sections that the performance in other cases may be significantly worse than that of the TBP.

2.5 Analysis of The Tandem Queue: General Case

In this section, we begin by analyzing the TBP for the tandem queueing network when $\mu_1 < \infty$.

Figure 2.3 illustrates the MC of the tandem queueing network under the TBP with $TH = 1$. Recall than under the TBP it is not possible to reach a state (q_1, q_2) such that $q_2 - q_1 > TH + 1$. As illustrated in the figure, the states can be classified into three groups, depending on whether customers waiting for service at Station 1 experience stoppage before they enter Station 2. For example, if the system is currently in state $(2, 0)$, neither customer at Station 1 can possibly experience any stoppages before entering Station 2. The same is true for all the other states above the dashed line in the top left corner of Figure 2.3. On the other hand, in all states to the right of the dashed boundary line, Station 1 is idled, thus all customers at this station will experience one or more stoppage before entering Station 2; state $(2, 3)$ is an example of this type.

Finally, customers at Station 1 in all the states below and to the left of the dashed line may or may not experience a stoppage before entering Station 2. Consider, for example, state $(3, 0)$. While the first two customers at Station 1 will not experience a stoppage, the situation is less clear for the last customer. We refer to this customer as "TC". If the next two events are both "Completion 1", the system will move to state $(1, 2)$ and TC will be stopped. If, on the other hand, at least one of the next two events is Arrival or Completion 2, the TC will not be stopped.

This discussion illustrates why the analysis of the TBPs is challenging. The number of

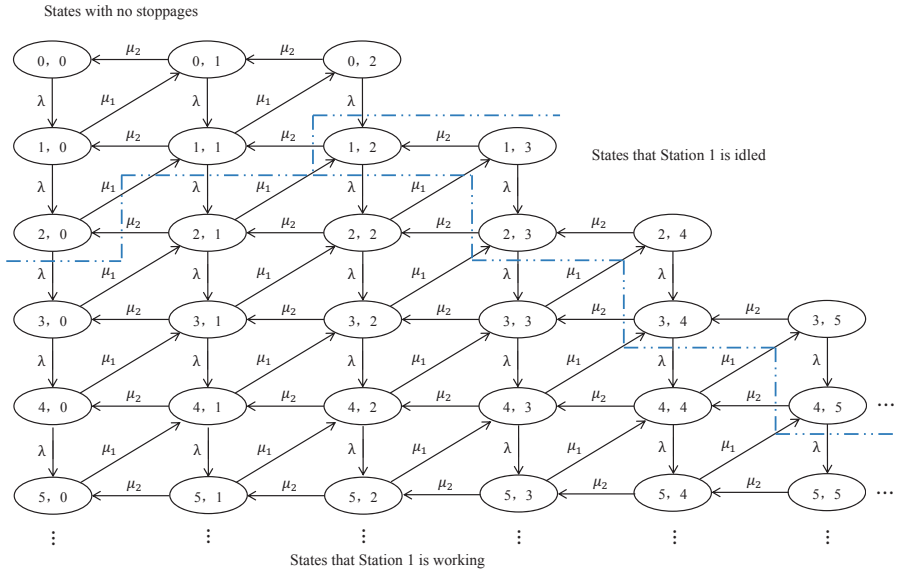

Figure 2.3: The MC (Q_1, Q_2) for $TH = 1$.

stoppages experienced by the TC (and thus the distribution of her waiting time) depends on queue lengths at both stations and on customers arriving after TC, i.e., this number depends on *future* events. The latter dependency prevents us from using Distributional Little's Law. Furthermore, the distribution of waiting time experienced by a customer depends not just on the state of the system, but also on the customer's position in the line at Station 1. As discussed in the example above, the customer immediately in front of TC will not experience any stoppages, and thus his distribution of the waiting time is clearly different from that for the TC. This implies that the observed state (q_1, q_2) of the system is not sufficient to uniquely express the distribution of $W_1^{q_1,q_2}$.

To overcome this difficulty, we augment the state space with a position indicator for each customer. Specifically, for each TC, in addition to the queue length indicators we also include the position of the TC in Station 1; we denote this position s for $s \geq 1$. Note that each TC now generates a new MC upon arrival, which we name TCMC.

This TCMC has three dimensions. When the TC arrives in state (q_1, q_2), she joins Station 1 and becomes the $s^{th} = (q_1+1)^{th}$ customer, so that the first state of TCMC is $(q_1 + 1, q_2, q_1 + 1)$. If we consider all states with the same s as one layer, each layer looks similar to the MC in Figure 2.3 except that there are no states with $q_1 > s$. The same three events discussed in

Section 2.3 may occur in the TCMC as well. Their effect on state (q_1, q_2, s) is as follows:

1. *Arrival*: TCMC transitions to state (q_1+1, q_2, s). Arrivals occur with rate λ in any state.

2. *Completion 1*: TCMC transitions to state $(q_1-1, q_2+1, s-1)$. This happens with rate μ_1, if $q_1 > 0$ and $\delta < TH$ (when Station 1 is not idled).

3. *Completion 2*: TCMC transitions to state (q_1, q_2-1, s). This event occurs with rate μ_2, if $q_2 \geq 1$.

When $s > 1$, the TC is waiting in Station 1. When $s = 1$, the TC is either in service or is the first in line to enter service when the stoppage of Station 1 ends. Since $\lambda < \mu_1$, the TCMC (q_1, q_2, s) will be absorbed in some state with $s = 0$, when the TC moves to Station 2. Let $X^{q_1,q_2,s}$ represent the TC's performance measure, given the network is in state (q_1, q_2, s).

To obtain the performance measure using (2.2) we can keep track of the TCMC starting from the state (q_1+1, q_2, q_1+1) and calculate the conditional performance measure according to all possible paths the TC may take until an absorbing state is reached. However, because the TCMC is three-dimensional, the required computational effort grows rapidly using this intuitive approach. We thus simplify the problem as shown below.

We first show that, similarly to the $\mu_1 = \infty$ case, the number of stoppages can be bounded.

Lemma 1 *If the TCMC is in state (q_1, q_2, s), then the maximum number of stoppages the TC may see, $M^{q_1,q_2,s}$, is*

$$M^{q_1,q_2,s} = \max\{2s - TH + \delta(q_1, q_2) - 1, 0\}.$$

Specially, if $\delta(q_1, q_2) \leq TH - 2s + 1$, there will be no stoppage for the TC. Thus, the performance measure experienced by a customer that reaches such states are independent of future arrivals.

It is easy to see that $\hat{M}^{q_1,q_2} = M^{q_1+1,q_2,q_1+1}$, i.e., Lemma 1 shows that, the number of stoppages the TC sees in the $\mu_1 = \infty$ case is the *maximum* number of stoppages the TC may see in $\mu_1 < \infty$ case. The reason is that when $\mu_1 = \infty$, the service time of Station 1 is zero, so the set of sequential events: Completion 1⇒Arrival(or Completion 2)⇒Arrival(or Completion 2) repeats for sure; when $\mu_1 < \infty$, this set of sequential events repeats only in the worst case.

We define *a no-stoppage state* to be a state in TCMC s.t. $\delta(q_1, q_2) \leq TH - 2s + 1$, i.e., $M^{q_1,q_2,s} = 0$. For example, consider again state $(3, 0)$ in the MC on Figure 2.3. As previously discussed, states $(3, 0, 1)$ and $(3, 0, 2)$ in the corresponding TCMC are no-stoppage. On the other hand, by Lemma 1, $M^{3,0,3} = 1$, so stoppage may occur in state $(3, 0, 3)$.

Observe that once the TCMC reaches a no-stoppage state, the network acts like a non-idling tandem queueing network for the TC and the distributions of the three steady state service measures can be calculated directly (see below). In the following sections, we treat no-stoppage states as absorbing states and use a recursion method to develop all three performance measures as follows:

- If state (q_1, q_2, s) is a no-stoppage state, i.e., $\delta \leq TH - 2s + 1$, then the distribution of $X^{q_1,q_2,s}$ can be calculated from Propositions 4 and 5 below.

- If Station 1 is stopped, i.e., $\delta = TH$ or $TH + 1$, both Arrival (w.p. $\frac{\lambda}{\lambda+\mu_2}$) and Completion 2 (w.p. $\frac{\mu_2}{\lambda+\mu_2}$) can happen in the TCMC. Using conditional probability, the distribution of $X^{q_1,q_2,s}$ can be recursively calculated from the distributions of $X^{q_1+1,q_2,s}$ and $X^{q_1,q_2-1,s}$.

- For states (q_1, q_2, s) such that $TH - 2s + 1 < \delta \leq TH - 1$, Arrival (w.p. $\frac{\lambda}{\lambda+\mu_1+\mu_2}$), Completion 1 (w.p. $\frac{\mu_1}{\lambda+\mu_1+\mu_2}$), and Completion 2 (w.p. $\frac{\mu_2}{\lambda+\mu_1+\mu_2}$) can all happen in the TCMC. Using conditional probability, the distribution of $X^{q_1,q_2,s}$ can be calculated from the distributions of $X^{q_1+1,q_2,s}$, $X^{q_1-1,q_2+1,s-1}$, and $X^{q_1,q_2-1,s}$.

Calculating the LT of $X^{q_1,q_2,s}$, similarly to (2.2), requires the steady state probability vector of the MC (Q_1, Q_2). It is easily seen that this MC is irreducible and aperiodic and has equilibrium probabilities, π_{q_1,q_2}. The balance equation for the (Q_1, Q_2) MC are (these are easier to follow when looking at Figure 2.3):

1) When $\delta < TH$ and $q_1 = q_2 = 0$, we have $\lambda\pi_{0,0} = \mu_2\pi_{0,1}$;

2) When $\delta < TH$ and $q_1 > 0$, $q_2 = 0$, we have $(\lambda + \mu_1)\pi_{q_1,0} = \lambda\pi_{q_1-1,0} + \mu_2\pi_{q_1,1}$;

3) When $\delta < TH$ and $q_1 > 0$, $q_2 > 0$, we have $(\lambda+\mu_1+\mu_2)\pi_{q_1,q_2} = \lambda\pi_{q_1-1,q_2} + \mu_1\pi_{q_1+1,q_2-1} + \mu_2\pi_{q_1,q_2+1}$;

4) When $\delta \leq TH$ and $q_1 = 0$, $0 < q_2 \leq TH$, we have $(\lambda + \mu_2)\pi_{0,q_2} = \mu_1\pi_{1,q_2-1} + \mu_2\pi_{0,q_2+1}$;

5) When $\delta = TH$ and $q_1 > 0$ (then $q_2 = q_1 + TH$), we have $(\lambda + \mu_2)\pi_{q_1,q_2} = \lambda\pi_{q_1-1,q_2} + \mu_1\pi_{q_1+1,q_2-1} + \mu_2\pi_{q_1,q_2+1}$;

6) When $\delta = TH + 1$ and $q_1 \geq 1$ (implying $q_2 = q_1 + TH + 1$), we have $(\lambda + \mu_2)\pi_{q_1,q_2} = \mu_1\pi_{q_1+1,q_2-1}$;

7) We also require $\sum_{q_1,q_2} \pi_{q_1,q_2} = 1$.

To solve these balance equations, we approximate π_{q_1,q_2} by assuming that Station 1 has a finite waiting room of size $Limit$. For any finite value of $Limit$, we can calculate an approximation of π_{q_1,q_2} by solving the balance equations numerically. When $Limit$ goes to infinity, the approximation approaches π_{q_1,q_2}. In our numerical experiments we found that $P\{q_1 = 100\} < 10^{-5}$, so $Limit = 100$ appears to be an adequate value for our parameter choices.

2.5.1 Distribution of Waiting Time for Station 1: W_1

In this section, we consider the TC's waiting time for Station 1, W_1. Note that there are two components of W_1: the time spent waiting for $s - 1$ Completion 1 events, and the time spent when Station 1 is idled. The first component depends only on s, and the second one is determined by s and $\delta = q_2 - q_1$. Thus, given s and δ, W_1 does not depend on the values of q_1 and q_2. Indeed, from Lemma 1 we see that the maximum number of stoppage $M^{q_1,q_2,s}$ only depends on s and δ; thus, we will next use $M^{s,\delta}$ to denote the maximum number of stoppages for a customer that is at a position s in queue 1 when $q_2 - q_1 = \delta$. A revised TCMC, with the state description (s, δ), is illustrated on Figure 2.4 for the case $TH = 1$; this simplified TCMC will be used to compute W_1.

Arrival or Completion 2 events do not affect s; these events only decrease the value of δ by 1. Completion 1 decreases the value of s by 1, and increase the value of δ by 2. If $\delta = TH$ or $TH + 1$, Station 1 is idled, so that the next event can only be Arrival or Completion 2.

In Figure 2.4, the column on the right-hand side, starting from $(1, 0)$, represents the no-stoppage states established in Lemma 1. The states above the doted line are states where Station 1 is idled, i.e., with $\delta \geq TH = 1$.

Let $W_1^{s,\delta}$ be the TC's waiting time for Station 1 while the network is in state (s, δ), and denote its LT by $L_{W_1^{s,\delta}}(h)$. $W_1^{s,\delta}$ is composed of two parts. The first part is the service time of the $s - 1$ customers in front of the TC in Station 1. This service time distribution is

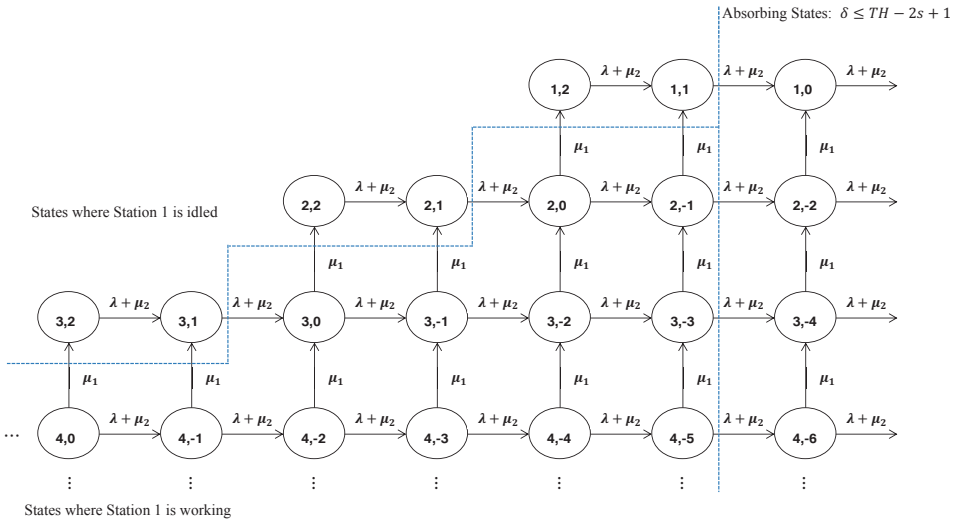

Figure 2.4: The revised TCMC (s, δ), when $TH = 1$.

$Erlang(s-1, \mu_1)$. The second part consists of stoppages in Station 1. As in the $\mu_1 = \infty$ case, the length of each stoppage is $\exp(\lambda + \mu_2)$.

Let $B_1^{s,\delta}$ denote the actual number of stoppages the TC will experience if she is in state (s, δ). Thus, the LT of $W_1^{s,\delta}$ is

$$L_{W_1^{s,\delta}}(h) = \left(\frac{\mu_1}{\mu_1 + h}\right)^{s-1} \sum_{i=1}^{M^{s,\delta}} P\left\{B_1^{s,\delta} = i\right\} \left(\frac{\lambda + \mu_2}{\lambda + \mu_2 + h}\right)^i, \tag{2.15}$$

where $M^{s,\delta}$ can be found from Lemma 1 and $\sum_{i=1}^{M^{s,\delta}} P\left\{B_1^{s,\delta} = i\right\} = 1$.

Thus, finding $L_{W_1^{s,\delta}}(h)$ is equivalent to finding the distribution of $B_1^{s,\delta}$, for any $s \geq 1$, $\delta \leq TH + 1$. This can be done as follows:

- If (s, δ) is a no-stoppage state, i.e., $\delta \leq TH - 2s + 1$, then $B_1^{s,\delta} = 0$ from Lemma 1.
- If Station 1 is stopped, i.e., for states with $\delta = TH$ or $TH + 1$, $B_1^{s,\delta}$ has the same distribution as $1 + B_1^{s,\delta-1}$.
- Otherwise, for states (s, δ) such that $TH - 2s + 1 < \delta \leq TH - 1$, the TCMC will go to state $(s, \delta - 1)$ (w.p. $\frac{\lambda+\mu_2}{\lambda+\mu_1+\mu_2}$), or to state $(s - 1, \delta + 2)$ (w.p. $\frac{\mu_1}{\lambda+\mu_1+\mu_2}$). Therefore, $B_1^{s,\delta}$ is distributed the same as $B_1^{s,\delta-1}$ or $B_1^{s-1,\delta+2}$, depending on which state the TCMC transitions to.

Since $s \in \{1, \ldots, Limit\}$ and $\delta \in \{-Limit, \ldots, TH+1\}$, the distribution of $B_1^{s,\delta}$ can now be computed iteratively; see Algorithm 1 in Appendix 2.9.2 for details.

2.5.2 Distribution of Waiting Time for Station 2: W_2

In this section we calculate $W_2^{q_1,q_2,s}$ – the TC's waiting time for Station 2, given that the network is at state (q_1, q_2, s). Let $K^{q_1,q_2,s}$ be the number of customers the TC sees when she enters Station 2. Given $K^{q_1,q_2,s} = k$, we know that $W_2^{q_1,q_2,s} \sim Erlang(\mu_2, k)$. So once we know the distribution of $K^{q_1,q_2,s}$, the LT of $W_2^{q_1,q_2,s}$ can be expressed as

$$L_{W_2^{q_1,q_2,s}}(h) = \sum_{k=0}^{q_2+s-1} (\frac{\mu_2}{\mu_2+h})^k P\left\{K^{q_1,q_2,s} = k\right\}. \tag{2.16}$$

We next derive the distribution of $K^{q_1,q_2,s}$, first for no-stoppage states and then for states with stoppages.

Distribution of W_2 at No-stoppage States

First, assume that the network is currently in a no-stoppage state, i.e., (q_1, q_2, s), and $\delta \leq TH - 2s + 1$. Given Lemma 1 Station 1 will not be idled before the TC enters Station 2. Thus, the arrival process does not affect the network, and we need to only consider the service processes of Stations 1 and 2. Still, it is possible for Station 2 to be *starved*, i.e., $q_2 = 0$, before the TC enters this station. We next discuss how to consider the starvation periods when calculating the distribution of $K^{q_1,q_2,s}$.

We represent the service operation of the TC by a Random Walk (RW) process in a two dimensional lattice graph, where the x and y axes represent the number of customers served by the first and second servers, respectively. Let the TC be the N^{th} arrival to the original tandem queue. Denote the total number of customers served by stations 1 and 2 before TC's arrival by X_N and Y_N, respectively. Note that $X_N \in [0, ..., N-1]$, $Y_N \in [0, ..., X_N]$ and $q_2 = X_N - Y_N$. The RW process is depicted on Figure 2.5. Obviously, the RW cannot go above the line $x = y$ (service 1 must finish before service 2). When Station 1 completes service the RW moves to the right, and when Station 2 completes service the RW moves up. Because both service completions are exponentially distributed, when both stations are busy, $P\{RW\ moves\ right\} = \frac{\mu_1}{\mu_1+\mu_2}$ and

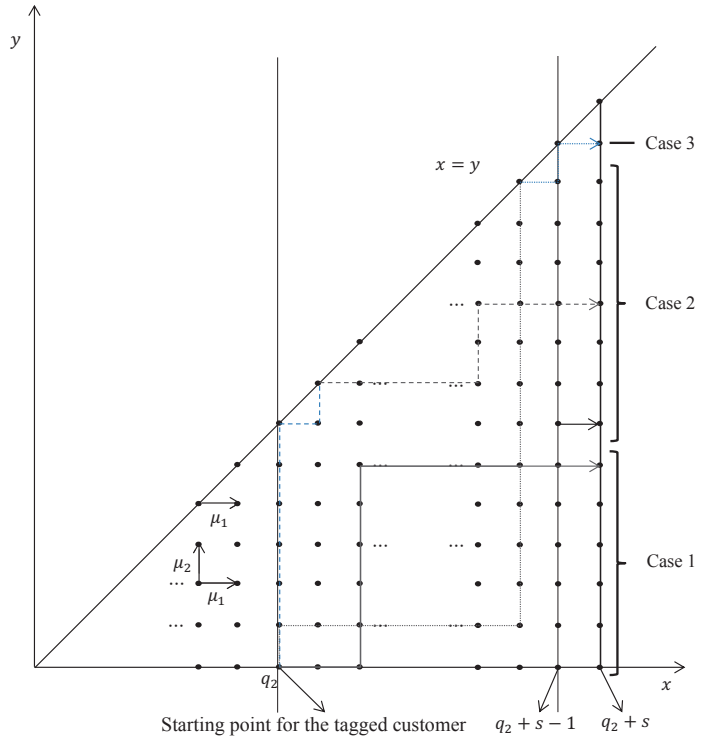

Figure 2.5: Lattice Graph of number of customers served by each station.

$P\{RW\ moves\ up\} = \frac{\mu_2}{\mu_1+\mu_2}$. Any point on the line $x = y$ means that Station 2 is starved and the next possible move for the RW is only to the right. We call points on the line $x = y$ points with Station 2 starved and other points in Figure 2.5 points with Station 2 working.

For any TC, we can ignore Y_N, because these customers have already left the network. Therefore, upon arrival of the TC we reset the starting point of the RW to $(X_N, Y_N) = (q_2, 0)$.

When the TC arrives to state (q_1, q_2, s) , there are q_2 customers in Station 2, which corresponds to the point $(q_2, 0)$ on Figure 2.5. When the TC finishes service in Station 1, this station has finished s customers, which represents the RW moving right s steps and reaching the line $x = q_2 + s$. By this time, Station 2 has served n customers, where $0 \leq n \leq q_2 + s - 1$. Thus, the sojourn time of TC at Station 1 corresponds to the time the RW moves from point $(q_2, 0)$ to a point on the line $(q_2 + s, n)$, with $0 \leq n \leq q_2 + s - 1$.

Let $B_2^{q_1,q_2,s}$, the number of times Station 2 is starved from when the TC arrives to the network and until she finishes service at Station 1. The joint distribution of n and $B_2^{q_1,q_2,s}$ can be calculated using the result from Milch and Waggoner (1970). This gives us the marginal

distribution of n. Since the number of customers TC sees upon entering Station 2 is $K^{q_1,q_2,s} = q_2 + s - 1 - n$, this also provides the distribution of $K^{q_1,q_2,s}$:

Proposition 4 *For any state (q_1, q_2, s) with $\delta \leq TH - 2s + 1$, the distribution of $K^{q_1,q_2,s}$ is:*

$$P\{K^{q_1,q_2,s} = k\} \tag{2.17}$$

$$= \begin{cases} \left[\binom{2s+q_2-3}{s-1} - \binom{2s+q_2-3}{s+q_2-1}\right]\left(\frac{\mu_1}{\mu_1+\mu_2}\right)^{s-1}\left(\frac{\mu_2}{\mu_1+\mu_2}\right)^{q_2+s-1} \\ \quad + \sum_{i=2}^{s}\left[\binom{2s+q_2-i-2}{s+q_2-2} - \binom{2s+q_2-i-2}{s+q_2-1}\right]\left(\frac{\mu_1}{\mu_1+\mu_2}\right)^{s-i}\left(\frac{\mu_2}{\mu_1+\mu_2}\right)^{q_2+s-1} & k = 0 \\ \left[\binom{2s+q_2-k-2}{s-1} - \binom{2s+q_2-k-2}{s+q_2-1}\right]\left(\frac{\mu_1}{\mu_1+\mu_2}\right)^{s}\left(\frac{\mu_2}{\mu_1+\mu_2}\right)^{q_2+s-k-1} \\ \quad + \sum_{i=1}^{s-k}\left[\binom{2s+q_2-k-i-2}{s+q_2-2} - \binom{2s+q_2-k-i-2}{s+q_2-1}\right]\left(\frac{\mu_1}{\mu_1+\mu_2}\right)^{s-i}\left(\frac{\mu_2}{\mu_1+\mu_2}\right)^{q_2+s-k-1} & 0 < k \leq s-1 \\ \binom{q_2+2s-2-k}{q_2+s-1-k}\left(\frac{\mu_1}{\mu_1+\mu_2}\right)^{s}\left(\frac{\mu_2}{\mu_1+\mu_2}\right)^{q_2+s-1-k} & s-1 < k \leq q_2+s-1. \end{cases} \tag{2.18}$$

Distribution of W_2 for States with Stoppages

To calculate $K^{q_1,q_2,s}$ for state with stoppage, observe:

- If Station 1 is stopped, i.e., $\delta = TH$ or $TH+1$, both Arrival (w.p. $\frac{\lambda}{\lambda+\mu_2}$) and Completion 2 (w.p. $\frac{\mu_2}{\lambda+\mu_2}$) can happen in the TCMC. So $K^{q_1,q_2,s}$ will be distributed as $K^{q_1+1,q_2,s}$ or $K^{q_1,q_2-1,s}$, depending on which event happens.

- For states (q_1, q_2, s) such that $TH - 2s + 1 < \delta \leq TH - 1$, Arrival (w.p. $\frac{\lambda}{\lambda+\mu_1+\mu_2}$), Completion 1 (w.p. $\frac{\mu_1}{\lambda+\mu_1+\mu_2}$), and Completion 2 (w.p. $\frac{\mu_2}{\lambda+\mu_1+\mu_2}$) can all happen in the TCMC. So $K^{q_1,q_2,s}$ will be distributed as $K^{q_1+1,q_2,s}$, $K^{q_1-1,q_2+1,s-1}$ and $K^{q_1,q_2-1,s}$, with these probabilities respectively.

Notice that the distribution of $K^{q_1,q_2,s}$ only depends on which no-stoppage state the process finally reaches, and is independent of the other details of the service process before that. Algorithm 2, given in the Appendix 2.9.2, uses these three conditions and Proposition 4 to express $K^{q_1,q_2,s}$ for any state (q_1, q_2, s). The distribution of $W_2^{q_1,q_2,s}$ can now be computed from (2.16).

Remark 3 *It may be of interest to compute the distribution of the total wait in the system for the TC, $W^{q_1,q_2,s} = W_1^{q_1,q_2,s} + W_2^{q_1,q_2,s}$. First note that $W_1^{q_1,q_2,s}$ and $W_2^{q_1,q_2,s}$ are not independent: since Station 2 is never intentionally idled, the longer the TC stays in Station 1, the less*

customers she will see, on average, when she enters Station 2. Still, in a similar way to Algorithms 1 and 2, one can calculate the distribution of $W^{q_1,q_2,s}$.

2.5.3 Distribution of Sojourn Time: S

In this section, we calculate the LT of sojourn time, which is the sum of waits and services in both stations, for the TC. This derivation allows us to express both $E[S]$ and $P\{S > t\}$.

We focus on Station 2. The TC's sojourn time, $S^{q_1,q_2,s}$, is between her arrival to the network and her departure, i.e., the time when Station 2 finishes serving $q_2 + s$ customers. We note that if there are customers in the network, Station 2 always serves customers when Station 1 is idled; and Station 1 always serves customers (if there are any customers in Station 1) when Station 2 is starved. Thus, the TC's sojourn time is composed of two parts. The first part is the service time of the $q_2 + s$ customers at Station 2, which is $Erlang(\mu_2, q_2 + s)$. The second part is the total time that Station 2 starves until it serves the TC. This time may depend on the behavior of the network after the TC's arrival and is therefore more challenging to characterize.

We know that the number of times Station 2 is starved $B_2^{q_1,q_2,s} \leq s$, because in the worst case Completion 2 happens q_2 times and then {Completion 1, Completion 2} sequence repeats until the TC is served at Station 2, so that $\sum_{i=0}^{s} P\{B_2^{q_1,q_2,s} = i\} = 1$.

Similar to (2.16), the common form of the LT of $S^{q_1,q_2,s}$ is

$$L_{S^{q_1,q_2,s}}(h) = \left(\frac{\mu_2}{\mu_2 + h}\right)^{q_2+s} \sum_{i=0}^{s} P\{B_2^{q_1,q_2,s} = i\} \left(\frac{\lambda + \mu_2}{\lambda + \mu_2 + h}\right)^{i}. \tag{2.19}$$

This transforms the problem to finding the distribution of $B_2^{q_1,q_2,s}$, for any state (q_1, q_2, s). We first consider no-stoppage states. As in the proof of Proposition 4, for the no-stoppage states we use the joint distribution of $q_2+s-1-K^{q_1,q_2,s}$ and $B_2^{q_1,q_2,s}$, $P\{q_2 + s - 1 - K^{q_1,q_2,s} = n, B_2^{q_1,q_2,s} = i\}$. Using this distribution and the Law of Total Probability, we get:

Proposition 5 *For any state (q_1, q_2, s) with $\delta \leq TH - 2s + 1$, the distribution of $B_2^{q_1,q_2,s}$ is*

$$P\left\{B_2^{q_1,q_2,s} = i\right\} = \begin{cases} \sum_{n=0}^{q_2-1} \binom{n+s-1}{n} \left(\frac{\mu_1}{\mu_1+\mu_2}\right)^s \left(\frac{\mu_2}{\mu_1+\mu_2}\right)^n \\ + \sum_{n=q_2}^{q_2+s-2} \left[\binom{s+n-1}{s-1} - \binom{s+n-1}{s+q_2-1}\right] \left(\frac{\mu_1}{\mu_1+\mu_2}\right)^s \left(\frac{\mu_2}{\mu_1+\mu_2}\right)^n, \ i = 0 \\ \sum_{n=q_2}^{q_2+s-2} \left[\binom{s+n-2}{s+q_2-2} - \binom{s+n-2}{s+q_2-1}\right] \left(\frac{\mu_1}{\mu_1+\mu_2}\right)^{s-1} \left(\frac{\mu_2}{\mu_1+\mu_2}\right)^n \\ + \left[\binom{2s+q_2-3}{s-1} - \binom{2s+q_2-3}{s+q_2-1}\right] \left(\frac{\mu_1}{\mu_1+\mu_2}\right)^{s-1} \left(\frac{\mu_2}{\mu_1+\mu_2}\right)^{q_2+s-1}, \ i = 1 \\ \sum_{n=q_2}^{q_2+s-2} \left[\binom{s+n-i-1}{s+q_2-2} - \binom{s+n-i-1}{s+q_2-1}\right] \left(\frac{\mu_1}{\mu_1+\mu_2}\right)^{s-i} \left(\frac{\mu_2}{\mu_1+\mu_2}\right)^n \\ + \left[\binom{2s+q_2-i-2}{s+q_2-2} - \binom{2s+q_2-i-2}{s+q_2-1}\right] \left(\frac{\mu_1}{\mu_1+\mu_2}\right)^{s-i} \left(\frac{\mu_2}{\mu_1+\mu_2}\right)^{q_2+s-1}, \ 2 \leq i < n - q_2 + 1 \end{cases} . \tag{2.20}$$

We can now calculate $B_2^{q_1,q_2,s}$ for any state (q_1, q_2, s) as follows:

- If the state (q_1, q_2, s) is in a no-stoppage state, i.e., $\delta \leq TH - 2s + 1$, the distribution is given by Proposition 5.
- If Station 1 is idled, i.e., $\delta = TH$ or $TH + 1$, both Arrival (w.p. $\frac{\lambda}{\lambda+\mu_2}$) and Completion 2 (w.p. $\frac{\mu_2}{\lambda+\mu_2}$) can happen in the TCMC. So $B_2^{q_1,q_2,s}$ will be distributed as $B_2^{q_1+1,q_2,s}$ or $B_2^{q_1,q_2-1,s}$.
- For states (q_1, q_2, s) such that $TH - 2s + 1 < \delta \leq TH - 1$ and $q_2 = 0$, there is no customer in Station 2. Arrival (w.p. $\frac{\lambda}{\lambda+\mu_1}$) and Completion 1 (w.p. $\frac{\mu_1}{\lambda+\mu_1}$) can happen in the TCMC. So $B_2^{q_1,0,s}$ is distributed as $B_2^{q_1+1,0,s}$ or $B_2^{q_1-1,1,s-1} + 1$.
- For states (q_1, q_2, s) such that $TH - 2s + 1 < \delta \leq TH - 1$ and $q_2 \neq 0$, Arrival (w.p. $\frac{\lambda}{\lambda+\mu_1+\mu_2}$), Completion 1 (w.p. $\frac{\mu_1}{\lambda+\mu_1+\mu_2}$) and Completion 2 (w.p. $\frac{\mu_2}{\lambda+\mu_1+\mu_2}$) can all happen in the TCMC. So $B_2^{q_1,q_2,s}$ will be distributed as $B_2^{q_1+1,q_2,s}$, $B_2^{q_1-1,q_2+1,s-1}$ or $B_2^{q_1,q_2-1,s}$.

Algorithm 3 in Appendix 2.9.2 use these four conditions to compute the distribution of $B_2^{q_1,q_2,s}$ for any state (q_1, q_2, s). The LT of the sojourn times $S^{q_1,q_2,s}$ can then be computed from (2.19).

2.6 Insights for $\mu_1 < \infty$ Case

In this section, we compare the performance of TBP, non-idling policy, and Kanban policies with respect to the expected sojourn time, $E[S]$, and the probability of excessive waits, $PW(t)$.

2.6.1 Insight 3: Comparing the TBP and Non-Idling Policy

First, we compare the performance of TBP and the non-idling policy. The key questions are: (1) what degree of improvement can be achieved by the TBP for the $PW(t)$ measure, and (2) by how much do sojourn times have to increase to achieve this improvement. We note that service measure $P\{S > t'\}$ could be used in place of $E[S]$. Numerical results show that the trade-off curves of $PW(t)$ and $P\{S > t'\}$ behave the same as the trade-off curves of $PW(t)$ and $E[S]$, so only $E[S]$ is considered in our numerical results.

The expressions for the service measures for the non-idling policy are determined by λ, μ_1, μ_2, and t, and can be obtained from e.g., using Burke's theorem (Burke, 1956):

$$E[S^{NI}] = \sum_{i=1}^{2} \frac{1}{\mu_i - \lambda}; \quad PW^{NI}(t) = \frac{1}{2} \sum_{i=1}^{2} \frac{\lambda}{\mu_i} e^{-(\mu_i - \lambda)t}.$$

To illustrate the trade-off between $PW^{TBP}(t)$ and the expected sojourn time under the TBP, $E\left(S^{TBP}\right)$ we proceed as follows. We initially set $\lambda = .85, \mu_1 = 1$ and $\mu_2 = .9$. Thus, Station 2 is the bottleneck, and the system utilization ratio $\rho = \rho_2 = .85/.9 \approx 94\%$. Next we select t such that $PW^{NI}(t) = 10\%$ - from the expressions above this value is $t = 31.78$ and $E[S^{NI}] = 26.67$.

We calculate the performance measures $E[S^{TBP}]$ and $PW^{TBP}(t)$ using $TH = 100, 99, \ldots, 0$. For $TH = 100$ the performance measures $(E[S^{TBP}], PW^{TBP}(31.78)) = (26.67, 0.1)$ are identical to these measures for the non-idle system. The results in Figure 2.6(a) present the trade-off curve of the TBP for different thresholds. The points corresponding to selected TH values are labeled on the curve (they decrease from left to right).

From the figure, we observe that the average sojourn times along the $x-$axis increase as TH values are decreased from 100: the lower the threshold the more the TBP departs from the non-idling policy, with the incidents of idling of Station 1 increasing. At $TH = 0$ the $E[S^{TBP}] = 30.5$ - a 14.4% increase over $E\left(S^{NI}\right)$, the expected sojourn time under the non-idle policy. Initially, as TH is decreased from 100, the $PW(t)$ values are reduced, indicating that the TBP is achieving the desired trade-off between the two performance measures. The $PW^{TBP}(t)$ is minimized at just over 7%, corresponding to $TH^* = 13$ (labeled with a star). For this TH

value, $E[S^{TBP}] = 27.31$. Thus, a TBP with $TH = 13$ achieves a nearly 30% improvement in the $PW(t)$ measure (7% vs 10%) at the cost of increasing the expected sojourn times by about 2% (from 26.67 to 27.31) - a trade-off that may be quite attractive. Reducing TH below 13 turns out to be counter-productive, thus, from the point of view of bi-objective optimization, the TH values below 13 are Pareto-inferior. However, all TH values greater than or equal to 13 are Pareto-optimal.

To gain additional insight, in Figure 2.6(b), we plot $P(W_i > 31.78)$ for $i = 1, 2$ under the TBP. Since Station 2 is the bottleneck in this case, the probability of wait longer than t is much greater there under the non-idling policy. This is shown on the extreme left of the plot where $TH = 100$ and the TBP is essentially identical to NI policy. For very high TH values most of the contribution to $PW(t)$ comes from Station 2. As TH is reduced, $P(W_1 > t)$ is increasing and $P(W_2 > t)$ is declining. Eventually, when TH decreases below 10, there is a much higher probability of long waits at Station 1 than at Station 2. It is interesting to note that $PW(t)$ is minimized at $TH^* = 13$ when the values of $P(W_1 > t)$ and $P(W_2 > t)$ are approximately equal. We have observed similar behavior with other parameter settings as well.

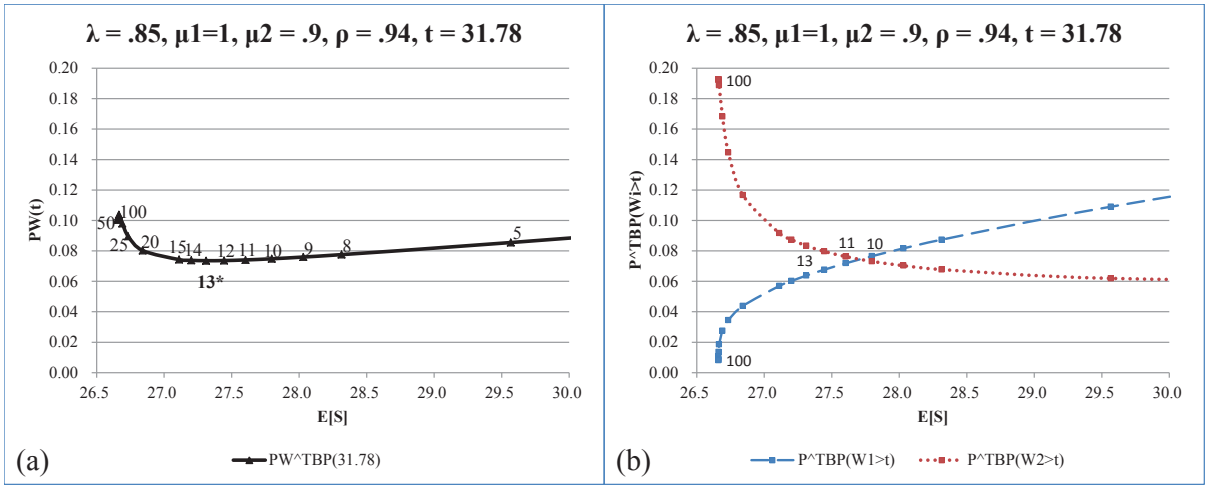

Figure 2.6: Trade-off curves corresponding to excessive wait probabilities of 10% (under non-idling policy). Non-idling policy corresponds to the left-most point on each curve.

We have observed from numerical results with different parameter settings that $P(W_1 > t)$ is a concave increasing function and $P(W_2 > t)$ is a convex decreasing function of $E[S]$, as on Figure 2.6(b). However, for different values of t, the behavior of $PW(t) = \frac{1}{2}(P(W_1 > t) + P(W_2 > t))$ as a function of $E[S]$ varies, typically being convex in some regions and concave in others.

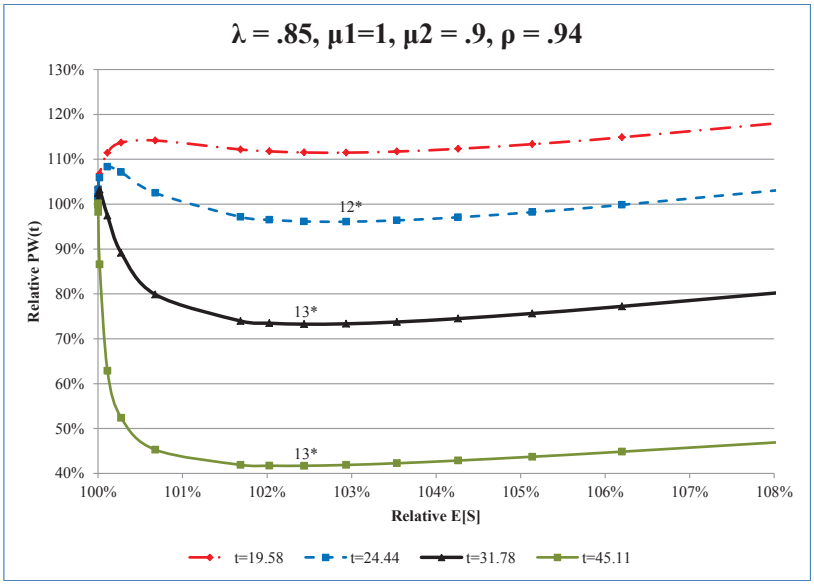

Figure 2.7: Trade-off curves of TBP corresponding to excessive wait probabilities of 20%, 15%, 10%, and 5% (under no-idling policy) for different system parameters. Non-idling policy corresponds to the left-most point on each curve.

To illustrate the improvements that can be achieved with TBP compared with the NI policy for different values of t, we plot the relative change in $PW(t)$ versus the relative change in $E[S]$ for four different values of t on Figure 2.7. Here 100% on both axes relates to corresponding values for the NI policy (or, equivalently, TBP(100) policy). Thus, on the $x-$axis the values increase from 100% since introducing SI can only hurt the expected service times, while on the $y-axis$ we have values above and below 100% since the TBP can improve or hurt the $PW(t)$ objective. The three values of $t = 45.11, 31.78, 24.44$, and 19.58 were selected to correspond to "excessive wait" probabilities of $5\%, 10\%, 15\%$, and 20% under the NI policy, respectively.

For the case where excessive waits are rare ($t = 45.11$), TBP provides very attractive trade-offs: decreasing $PW(t)$ by close to 60% at the cost of increasing $E[S]$ by just 2%. Moreover, most of the decrease in $PW(t)$ occurs for even smaller values of $E[S]$, corresponding to thresholds higher than the $PW(t)$-minimizing value of $TH^* = 13$. Thus, the value of TH that minimizes $PW(t)$ may not be the best choice. The reduction in $PW(t)$ provided by TBP for $t = 31.78$ case is a bit smaller, but is also quite substantial at nearly 30%, while the increase in $E[S]$ is just over 2%.

The TBP is much less successful for the $t = 24.44$ case where "excessive waits" occur 15%

of the time under the NI policy. Here, as the threshold is decreased from 100, both objectives are initially hurt, with $PW(t)$ rising sharply. This is because the decrease in $P(W_2 > t)$ is very small, while $P(W_1 > t)$ increases rapidly. For lower TH values, the $PW(t)$ begins to fall, eventually falling about 5% below the value for the NI policy around $TH^* = 12$. The cost of this improvement is the 3% increase in $E[S]$. Thus, the trade-offs offered by the TBP are much less attractive in this case. We also observe that here $PW(t)$ is not a convex function of $E[S]$.

As the probability of excessive waits is increased to 20%, the Pareto-optimal trade-offs disappear: while the behavior of $PW(t)$ as TH values are increased is similar to the previous case (first an increase, then a slight decrease, followed by another increase), the level never gets below the value achieved for $TH = 100$, i.e., the value for the NI policy.

Thus, we observe similar patterns to the ones derived analytically for the asymptotic $\mu_1 = \infty$ case: the TBP reduces $PW(t)$ when the "excessive waits" are sufficiently rare in the system.

Since the TBP redistributes some waiting times from Station 2 to Station 1, intuitively it should be most effective when Station 2 is the system's bottleneck. This intuition is supported by Figure 2.8. The four curves presented on four panels correspond to t^* (dashed line) and values of t such that the probabilities of long waits are 1% (lower solid), 5% (middle solid), and 10% (top solid) under the NI policy. Figure 2.8(a-b) present results for cases where the processing rates of Stations 1 and 2 are identical. Figure 2.8(c-d) present cases where the processing rate of Station 2 is reduced to .95, making it more of a bottleneck. We see similar patterns to those described for the previous figure: the TBP reduces the $PW(t)$ in all cases at the cost of a small increase in $E[S]$; the relative improvement in $PW(t)$ is increasing in t. Moreover, we see that the improvements provided by TBP is greater when Station 2 is more of a bottleneck (Figure 2.8(a) v.s. (c) and (b) v.s. (d)); even under similar utilization levels but different arrival rates (Figure 2.8(b) v.s. (c)).

Figures 2.7 and 2.8 provide some intuitions on identifying t^*s and TH^*s for different parameter settings. We notice that TH^* is relatively stable for similar arrival rates, and that $PW^{NI}(t^*)$ is relatively stable for similar utilization level at Station 2. Specifically, comparing Figure 2.7 with Figure 2.8(a,c), $TH^* \in [11, 15]$ is stable for the same arrival rate, $\lambda = .85$. This is also supported by comparing Figure 2.8(b,d), where $TH^* \in [16, 23]$. Similarly, $PW^{NI}(t^*)$ is stable under similar utilization levels. For example, when $\rho = .95$ (Figure 2.8(d) and Figure

2.7), $PW^{NI}(t^*)$ is about 15%; and when $\rho = .9$ (Figure 2.8(b) and (c)), $PW^{NI}(t^*)$ is about 12%.

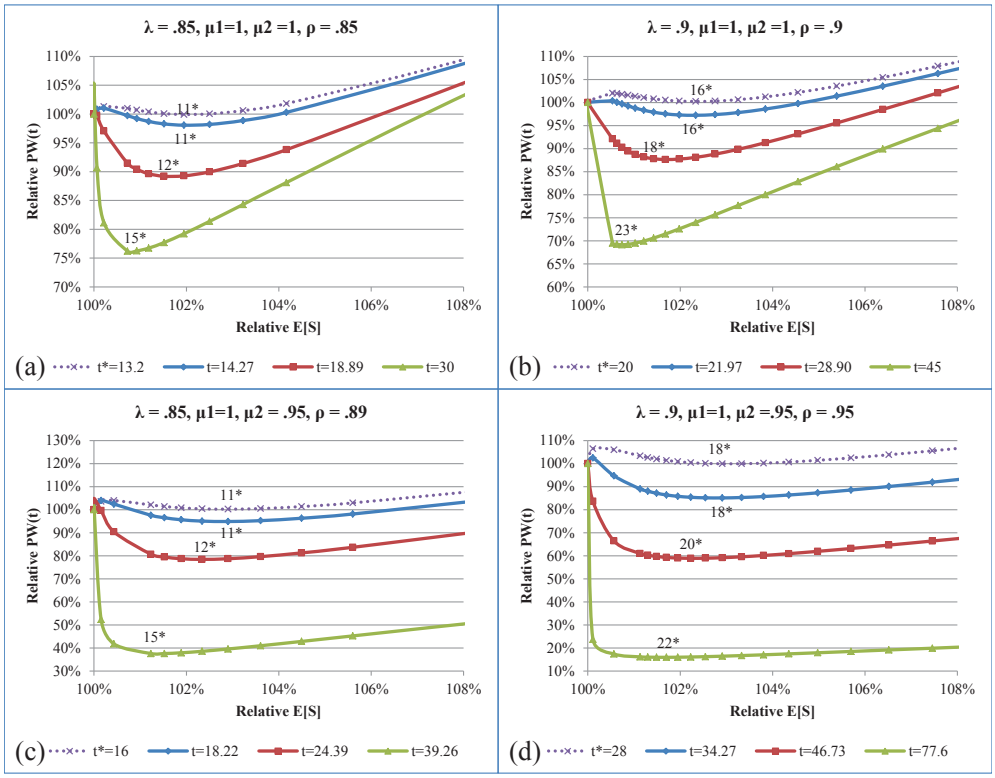

Figure 2.8: Trade-off curves of TBP corresponding to t^* and excessive wait probabilities of 10%, 5% and 1% (under non-idling policy) for different system parameters. Non-idling policy corresponds to the left-most point on each curve.

2.6.2 Insight 4: Comparing the TBP and Kanban policies

As the analytical derivation of the waiting time at each station is not available and is beyond the scope of this paper, to compare the performance of Kanban and TBP, we constructed a simulation model using MATLAB. We simulated one million customers under the Kanban policies with $BS = 100, 99, ..., 1$ and the TBPs with $TH = 100, 99, ..., 1$. (Despite having analytic results for the TBP we use simulation so that we compare both policies under the same sample path.) The results are presented on Figure 2.9 for a system with $\mu_2 = .9$ on the left panel and $\mu_2 = .95$ on the right. In both cases the value of t was chosen to correspond to 10% probability of long wait under the NI policy. With $BS = 100$, the Kanban policy performs identically to the NI one, which gives us the starting point on each panel. We then decrease

the value of the buffer size BS in steps of 1 and plot the values of $PW(t)$ and $E[S]$ for each BS. We plot the TBP curve in a similar fashion.

First consider the panel on the left. While the TBP generally outperforms the Kanban policy (recall that the Pareto-optimal points are the ones on the south-western frontier), when Relative $E[S] \geq 101.54$, the Kanban policy outperforms the TBP, achieving lower $PW(t)$ values for the same sojourn times. We note that selecting the right BS value is very important - values that are too high or too low may lead to performance worse than the NI policy. In fact, our numerical experiments show that the BS^* that minimizes $PW(t)$ appears to be very sensitive to t, while the TBP is much more robust in this respect (see Table 2.1). This lack of robustness presents a challenge for implementing Kanban policies, as the exact value of t may differ among customers.

Now consider the right panel, where $\mu_2 = .95$. Here the TBP clearly dominates Kanban (which produces very few Pareto-optimal values). The intuition behind poor performance of the Kanban policy in this case is that Kanban policy ignores the queue size in front of Station 1. While this is not a major issue when Station 2 is the main bottleneck in the system (as on the left panel), when the processing rates of Stations 1 and 2 are similar (as on the right panel) and Station 1 is idled even when facing a long queue, long wait times occur. Thus, while Kanban policy performed very well for the asymptotic $\mu_1 = \infty$ case, the performance under more realistic conditions appears to be significantly worse. The additional flexibility afforded by the TBP, which takes both q_1 and q_2 into account, is apparently important in case of a more balanced system.

t	TH^*	$PW^{TBP(TH^*)}(t)$	$E^{TBP(TH^*)}[S]$	BS^*	$PW^{Kanban(BS^*)}(t)$	$E^{Kanban(BS^*)}[S]$	$PW^{NI}(t)$	$E^{NI}[S]$
15	100	0.2663	26.20	100	0.2664	26.20	0.2662	26.20
20	100	0.1930	26.20	100	0.1931	26.20	0.1930	26.20
25	12	0.1329	26.86	22	0.1243	27.76	0.1437	26.20
30	12	0.0810	26.86	25	0.0764	26.99	0.1082	26.20
35	12	0.0485	26.86	27	0.0470	26.71	0.0828	26.20
40	13	0.0284	26.74	31	0.0293	26.41	0.0630	26.20
45	12	0.0158	26.86	34	0.0180	26.30	0.0482	26.20
50	11	0.0079	27.00	37	0.0109	26.24	0.0371	26.20

Table 2.1: Performance of TBP and Kanban policies for different values of t.

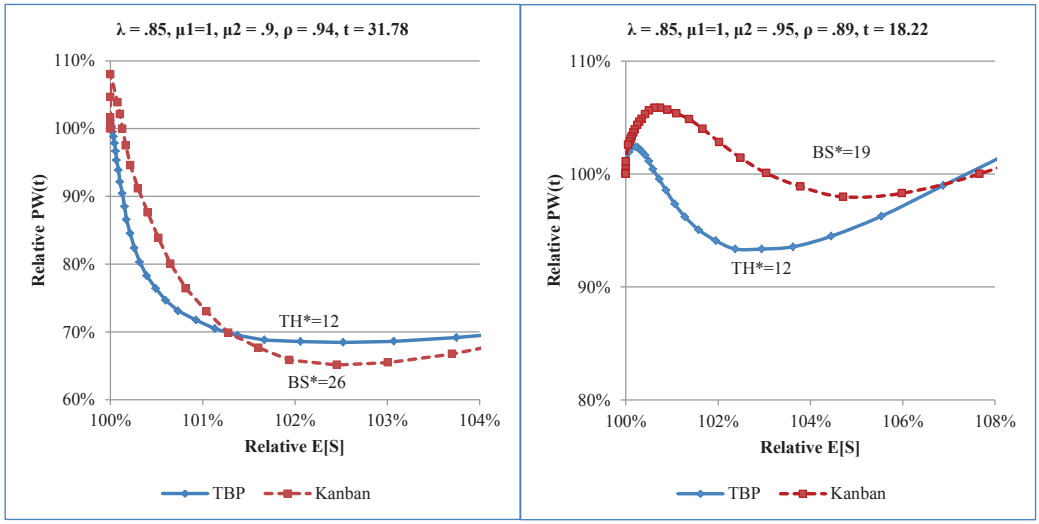

Figure 2.9: Trade-off curves corresponding to $t = 31.78$ under the TBP and the Kanban policy.

2.7 Summary and Open Questions

In this paper we studied strategic idling - i.e., purposefully idling some upstream stations when the downstream stations become too busy, in a two-station tandem queue network. The purpose of SI is to reduce the incidence of excessive waits and thus improve customer service experience in queueing networks. Numerical results indicate that TBP can be quite effective in reducing the incidence of excessive waits, without significantly increasing system sojourn times. Thus, TBP makes it possible to improve the service experience of customers without adding any capacity to the system (by, instead, idling some of the existing capacity). A comparison with Kanban policies indicates that the TBP is more efficient.

We demonstrated that these insights hold in more general settings. Specifically, in Appendix 2.9.3, we present a simple example that illustrates possible TBPs and Kanban policies for a 3-station serial queueing network with exponential service time at each station and Poisson arrivals; and in our working paper, Baron et al. (2013), we consider an open-shop queueing network that does not reach steady state. Both studies used simulation. The results indicate that the managerial insights listed earlier for the 2-station system likely hold in other more general settings as well. A generalization of TBP to n-station tandem queue system is presented at Appendix 2.9.4.

Clearly, this paper undertakes only an initial study of the TBPs and SI and much work

remains to be done. It would be interesting to inspect the effect of the TBP in an Emergency Department setting and compare the result with Saghafian et al. (2012). It would be very beneficial to extend our analytical results to more general settings (n-station networks, non-stationary arrival rate, general service time etc.), though this appears to be quite difficult. In particular, the structure of the optimal TBPs (i.e., the specification of δ functions and the TH values) needs to be investigated.

There are several other possible directions for future research. An analysis of waiting time distributions under either of the control policies developed for manufacturing settings is an obvious one. It would also be interesting to further investigate the application of TBP and other policies with SI in additional settings such as open-shop queueing networks. Also, the trade-offs between other service level measures can be explored. Finally, in practice there is value to adequately define excessive wait and acceptable average sojourn times. Both measures should be related to customers patience and may be evaluated using customers' surveys.

2.8 References

Afèche, P. (2013) Incentive-compatible revenue management in queueing systems: optimal strategic delay. *Manufacturing Service Oper. Management* Forthcoming.

Baron, O., O. Berman, D. Krass. (2008) Facility Location with Stochastic Demand and Constraints on Waiting Time. *Manufacturing Service Oper. Management* 10(3) 484-505.

Baron, O., O. Berman, D. Krass, J. Wang (2013) Dynamic Scheduling and Strategic Idling in an Open-shop Queueing Network: Case Study and Analysis. Working Paper, Rotman School of Management, Toronto, Canada.

Baron, O., J. Milner. (2009) Staffing to Maximize Profit for Call Centers with Alternate Service Level Agreements. *Oper. Res.*, 57, pp. 685-700.

Bertsimas, D., G. Mourtzinou (1996) A unified method to analyze overtake free systems. *Adv. Appl. Probab.*, 28, 588-625.

Bertsimas, D., D. Nakazato, (1995) The Distributional Little's Law and Its Applications. *Oper. Res.* 43(2) 298-310.

Burke, P. (1956) The Output of a Queueing system. *Oper. Res.* 4(6) 699-704.

Caldentey, R., L. Wein. (2006) Revenue Management of a Make-to-Stock Queue. *Oper. Res.* 54(5), 859-875.

Chen H, D. D. Yao. (2001) Fundamentals of Queueing Networks, Performance, Asymptotics, and Optimization. *Springer*, New-York.

Conway, R., W. Maxwell, J.O. McClain, L.J. Thomas. (1988) The Role of Work-in-Process Inventory in Serial Production Lines. *Oper. Res.*, 36, 229-241.

de-Véricourt F., Y.-P. Zhou (2005) A routing problem for call centers with customer callbacks after service failure. *Oper. Res.* 53(6) 968-981.

de-Véricourt F., O. Jennings. (2011) Review on Nurse Staffing in Medical Units: A Queueing Perspective. *Oper. Res.* 59(6) 1320-1331, 1547-1548.

Friedman, H.H., Friedman, L.W. (1997) Reducing the "wait" in waiting-line systems: waiting line segmentation. *Business Horizons* 40(4): 54-58.

Gans, N., G. Koole, A. Mandelbaum. (2003) Telephone Call Centers: Tutorial, Review, and Research Prospects. *Manufacturing & service Oper. Management* 5(2): 79-141

Govil, M.K., M.C. Fu. (1999) Queueing Theory in Manufacturing: A Survey. *J. Manufacturing Systems*, 18(3) 214-240.

Ha A. (1997a) Inventory rationing in a make-to-stock production system with several demand classes and lost sales. *Management Sci.* 43(8) 1093-1103.

Ha A. (1997b) Stock-rationing policy for a make-to-stock production system with two priority classes and backordering. *Naval Res. Logist.* 44(5) 457-472.

Larson R. C. (1987) Perspectives on Queues: Social Justice and the Psychology of Queueing. *Oper. Res.* 35(6) 895-905.

Masin, M., Y. Herer, E. M. Dar-el. (2010) SWIP: A Unified Model of Self-regulating Production Control Systems. Working paper, Technion.

Mehrotra, V., K. Ross, G. Ryder, Y.P. Zhou. (2012) Routing to Manage Resolution and Waiting Time in Call Centers with Heterogeneous Servers. *Manufacturing & Service Oper. Management* 14(1) 66-81.

Milch P.R., M.H. Waggoner (1970) A Random Walk Approach to a Shutdown Queueing System. *Appl. Math.* 19(1), July.

Ross S.M. (2000) Introduction to Probability Models, Seventh Edition, *Academic Press.*

Saghafian S., W.J. Hopp, M.P. Van Oyen, J.S. Desmond, S.L. Kronick (2012) Patient Streaming as a Mechanism for Improving Responsiveness in Emergency Departments. *Oper. Res.* Forthcoming.

Sloberg, J.J. (1977) A Mathematical Model of Computerized Manufacturing Systems, *4th Interna. Conf. on Production Res. (Tokyo)*, Pre-Print 8.4.

Soman, D., M. Shi (2003) Virtual Progress: The effect of path characteristics on perceptions of progress and choice. *Management Sci.* 49(9) 1229-1250.

Spearman, M., D. L. Woodruff, W. J. Hopp. (1990) CONWIP: a pull alternative to Kanban. *Internat. J. Production Res.* 28(5) 879-894.

Sugimori, Y., K. Kusunoki, F. Cho, S. Uchikawa. (1977) Toyota production system and Kanban system Materialization of just-in-time and respect-for-human system. *Int. J. Prod. Res.* 15, 553-564.

Taylor, S. (1994) Waiting for service: the relationship between delays and evaluation of service. *J. Marketing*, 58 (April), 56-69.

van Ryzin, G., S.X.C. Lou, S.B. Gershwin. (1993) Production control for a tandem two-machine system. *IIE Trans.*, 25(5), 5-20.

Veatch, M.H., L.M. Wein. (1994) Optimal Control of a Two-station Tandem Production/Inventory System. *Oper. Res.*, 42(2), 337-350.

Weber, R.R., S. Stidham Jr. (1987) Optimal Control of Service Rates in Networks of Queues. *Adv. Appl. Probab.*, 19(1), 202-218.

2.9 Appendix

2.9.1 Proofs

Proof of Proposition 1

From Figure 2.1 we have two observations. If the TC arrives at a state (q_1, q_2) s.t. $q_1+q_2 \leq TH$, then $q_1 = 0$. The TC enters Station 2 without being stopped in Station 1, i.e., $\hat{M}^{q_1,q_2} = 0$. If the TC arrives at a state (q_1, q_2) with $q_1 > 0$ then $q_2 = q_1 + TH$ or $q_2 = q_1 + TH + 1$, and the TC is stopped in Station 1. By Remark 2, every Completion 2 event will be followed by two

Arrival /Completion 2 events. Moreover, every time Arrival or Completion 2 event occurs, the value of δ changes, thus increasing the number of stoppages experienced by TC by 1. Therefore, the number of stoppage until the TC enters Station 2 will be:

$$\begin{aligned}\hat{M}^{q_1,q_2} &= \begin{cases} 2q_1 & \text{if } \delta = TH \\ 2q_1+1 & \text{if } \delta = TH+1 \end{cases} \\ &= q_1+q_2-TH.\end{aligned}$$

Proof of Proposition 2

We first denote the set of zero points of any discrete function $f(z)$ as $\{z \mid f(z) f(z+1) \leq 0\}$, i.e., zero points of $f(z)$ are those points where $f(z)$ switches sign. Let z^* denote a zero point of the forward difference of $f(z)$ denote as $f(z+1)-f(z)$. Using analogy between the forward difference of a discrete function and the first order derivative of a continuous function, we know z^* is the point where extremum of $f(z)$ may be reached. Specifically, if $f(z^*+1)-f(z^*) \geq 0$ and $f(z^*+2)-f(z^*+1) \leq 0$, then z^* is a local maximum point of $f(z)$; if $f(z^*+1)-f(z^*) \leq 0$ and $f(z^*+2)-f(z^*+1) \geq 0$, then z^* is a local minimum point of $f(z)$.

To find the minimum points of $PW^{TBP(TH)}(t)$, we investigate the zero points of the forward difference of $PW^{TBP(TH)}(t)$: $g_1'(t,TH) = PW^{TBP(TH+1)}(t) - PW^{TBP(TH)}(t)$.

We know from (2.12) that $g_1'(t,0)=0$, so $TH=0$ is a zero point of $g_1'(t,0)$. Furthermore, we know that $\lim_{TH\to\infty} PW^{TBP(TH)}(t) = PW^{NI}(t)$. Therefore, if we can show that $g_1'(t,TH)$ has either no zero points in $[1,\infty)$ or one zero point in $[1,\infty)$ which is a local maximum point of $PW^{TBP(TH)}(t)$, we can conclude that the minimum point of $PW^{TBP(TH)}(t)$ can only be at $TH=0$ or ∞.

Using (2.12) we write $g_1'(t,TH)$ for $TH \geq 1$:

$$g_1'(t,TH) = \begin{cases} \frac{1}{2}(1-\rho_2) e^{-(\mu_2-\lambda\rho_2)t}\left(1-e^{-\lambda\rho_2 t}-\rho_2^2\right) & if\ TH=1 \\ (1-\rho_2)\rho_2^{1-TH} e^{-(\mu_2-\lambda\rho_2)t}\left(1-e^{-\lambda\rho_2 t}\sum_{k=0}^{TH-1}\frac{(\lambda\rho_2 t)^k}{k!}-\rho_2^{2TH}\right) & if\ TH\geq 2 \end{cases}.$$

Denote $g_2(t,TH) = 1-e^{-\lambda\rho_2 t}\sum_{k=0}^{TH-1}\frac{(\lambda\rho_2 t)^k}{k!}-\rho_2^{2TH}$ for $TH \geq 1$, so that $g_1'(t,TH)$ has the

same zero points as $g_2(t, TH)$.

It is obvious that $1 - e^{-\lambda\rho_2 t}\sum_{k=0}^{TH-1}\frac{(\lambda\rho_2 t)^k}{k!} > 0$ and $\rho_2^{2TH} > 0$. We state without proof that $\lim_{TH\to\infty}\frac{1-e^{-\lambda\rho_2 t}\sum_{k=0}^{TH-1}\frac{(\lambda\rho_2 t)^k}{k!}}{\rho_2^{2TH}} = 0$, i.e., $1 - e^{-\lambda\rho_2 t}\sum_{k=0}^{TH-1}\frac{(\lambda\rho_2 t)^k}{k!}$ converges to zero faster than ρ_2^{2TH}. Thus, $\lim_{TH\to\infty} g_2(t, TH) = 0^-$, i.e., $g_2(t, TH)$ converges to zero from below (so does $g_1'(t, TH)$), when $TH \to \infty$. Thus $PW^{TBP(TH)}(t)$ decreases with TH when $TH \to \infty$ for any t.

Next, to find the zero points of $g_2(t, TH)$, we investigate the zero points of the forward difference of $g_2(t, TH)$:

$$g_2'(t, TH) = -\left(\frac{(\mu_2 t)^{TH}}{TH!} - \left(1 - \rho_2^2\right)e^{\lambda\rho_2 t}\right)e^{-\lambda\rho_2 t}\rho_2^{2TH}. \tag{2.21}$$

Denote $g_3(t, TH) = -\left(\frac{(\mu_2 t)^{TH}}{TH!} - \left(1 - \rho_2^2\right)e^{\lambda\rho_2 t}\right)$, so that $g_2'(t, TH)$ has the same zero points as $g_3(t, TH)$. Because $\frac{x^{TH}}{TH!}$ is a bell shape function, $g_3(t, TH)$ has at most two zero points.

When $g_2(t, 1) \neq 0$, we prove by contradiction that $g_2(t, TH)$ has either no zero points in $[1, \infty)$ or one zero point in $[1, \infty)$ which is a local maximum point of $PW^{TBP(TH)}(t)$:

- For the case when $g_2(t, 1) > 0$, assume that $g_2(t, TH)$ has two or more zero points in $[1, \infty)$. Because $\lim_{TH\to\infty} g_2(t, TH) = 0^-$, we know that $g_2(t, TH)$ switches sign for at least three times in $[1, \infty)$, i.e., $g_2(t, TH)$ has at least three zero points in $[1, \infty)$. Therefore, $g_2(t, TH)$ have at least three local extremum points in $[1, \infty)$, i.e., $g_2'(t, TH)$ have at least three zero points in $[1, \infty)$. This conflicts with that $g_3(t, TH)$ has at most two zero points. Therefore, $g_2(t, TH)$ has one zero point in $[1, \infty)$ and $g_2(t, TH)$ switches sign from positive to negative at this point. This point is thus a local maximum point of $PW^{TBP(TH)}(t)$.

- For the case when $g_2(t, 1) < 0$ (i.e., $\left(1 - \rho_2^2\right)e^{\lambda\rho_2 t} < 1$), we consider two sub-cases:

  1. For $g_2'(t, 1) < 0$, assume that $g_2(t, TH)$ has two or more zero points in $[1, \infty)$. Then, using a similar discussion as the previous bullet point, we know that $g_2'(t, TH)$ have at least three zero points. This conflicts with the fact that $g_3(t, TH)$ has at most two zero points. Therefore, $g_2(t, TH)$ has less than two zero points. Because $g_2(t, 1) < 0$

and $\lim_{TH\to\infty} g_2(t,TH) = 0^-$ are both below zero, we can conclude that $g_2(t,TH)$ has no zero points.

2. For $g_2'(t,1) \geq 0$ (i.e., $\mu_2 t \leq \left(1-\rho_2^2\right) e^{\lambda\rho_2 t}$), we have $\mu_2 t \leq \left(1-\rho_2^2\right) e^{\lambda\rho_2 t} < 1$. Let $g_3'(t,TH)$ be the forward difference of $g_3(t,TH)$, i.e., $g_3'(t,TH) = \frac{(\mu_2 t)^{TH}}{TH!}\frac{(TH+1)-\mu_2 t}{TH+1}$. Because $TH \geq 1$, we have $TH+1 > \mu_2 t$, so $g_3'(t,TH) > 0$ for any $TH \geq 1$. Using $g_3(t,1) > 0$ (because $g_2'(t,1) > 0$), we get $g_3(t,TH) > 0$ (i.e., $g_2'(t,TH) > 0$) for any $TH \geq 1$, i.e., $g_2(t,TH)$ is a monotone increasing function in $[1,\infty)$. Then, from $\lim_{TH\to\infty} g_2(t,TH) = 0^-$, we know that $g_2(t,TH)$ has no zero points in $[1,\infty)$.

To conclude, $g_2(t,TH)$ has either no zero points in $[1,\infty)$ or one zero point in $[1,\infty)$ which is a local maximum point of $PW^{TBP(TH)}(t)$.

For the case when $g_2(t,1) = 0$, we have $PW^{TBP(0)}(t) = PW^{TBP(1)}(t) = PW^{TBP(2)}(t)$, then the same discussion as $g_2(t,1) \neq 0$ case on $g_2(t,TH)$'s zero points in $[2,\infty)$ leads to the same conclusion.

Then, solving $PW^{TBP(0)}(t) = PW^{NI}(t)$ gives t^*.

Proof of Proposition 3

The stead-state distribution of the MC of the system under Kanban(BS) is the same as the distribution under $TBP(TH)$, i.e., π_{q_1,q_2} is given in (2.3).

Let $\tilde{W}_i^{q_1,q_2}$ denote the TC's waiting time at Station i $(i=1,2)$, given she arrives at state (q_1,q_2):

- If $q_1+q_2 \leq BS-1$, then no one is waiting for Station 1 and TC directly enters the waiting room for Station 2. Therefore, $\tilde{W}_1^{q_1,q_2} = 0$. Her waiting time for Station 2 is distributed as $Erlang(\mu_2, q_1+q_2)$. Of course, when $q_1+q_2=0$, TC does not wait for Station 2.

- if $q_1+q_2 \geq BS$, then there are q_1+q_2-BS customers waiting for Station 1 and BS customers waiting for Station 2. The TC should wait for Station 1 first. When the number of customers in Station 2 reduces to $BS-1$, she can enter Station 2's waiting room where $BS-1$ customers are waiting there. Therefore, TC's waiting time for Station

1 is distributed as $Erlang\,(\mu_2, q_1+q_2-BS+1)$ and her waiting time for Station 2 is distributed as $Erlang\,(\mu_2, BS-1)$.

Thus, using (2.2), (2.3) and the above discussion, we can now write the LT of $\tilde{W}_1$:

$$\begin{aligned} L_{\tilde{W}_1}(h) &= \sum_{i=0}^{BS-1}(1-\rho_2)\,\rho_2^i + \sum_{i=BS}^{\infty}(1-\rho_2)\,\rho_2^i\left(\frac{\mu_2}{\mu_2+h}\right)^{i-BS+1} \\ &= 1-\rho_2^{BS}+\rho_2^{BS}\frac{\mu_2-\lambda}{\mu_2-\lambda+h}. \end{aligned}$$

From the transform of $\tilde{W}_1$ we conclude that there is no waiting in Station 1 w.p. $1-\rho_2^{BS}$, and the waiting is distributed as an $\exp(\mu_2-\lambda)$ R.V. w.p. ρ_2^{BS}. Hence,

$$P\left\{\tilde{W}_1 > t\right\} = \rho_2^{BS} e^{-(\mu_2-\lambda)t}. \tag{2.22}$$

In a similar fashion, we get the LT of $\tilde{W}_2$, the TC's waiting time at Station 2,

$$\begin{aligned} L_{\tilde{W}_2}(h) &= 1-\rho_2+\sum_{i=1}^{BS-1}(1-\rho_2)\,\rho_2^i\left(\frac{\mu_2}{\mu_2+h}\right)^i + \sum_{i=BS}^{\infty}(1-\rho_2)\,\rho_2^i\left(\frac{\mu_2}{\mu_2+h}\right)^{BS-1} \\ &= 1-\rho_2+(1-\rho_2)\sum_{i=1}^{BS-1}\rho_2^i\left(\frac{\mu_2}{\mu_2+h}\right)^i + \rho_2^{BS}\left(\frac{\mu_2}{\mu_2+h}\right)^{BS-1}. \end{aligned}$$

From the LT of $\tilde{W}_2$ we know that there is no waiting in Station 2 w.p. $1-\rho_2$; the waiting is distributed as a $Erlang\,(\mu_2, i)$ R.V. w.p. $(1-\rho_2)\,\rho_2^i$ for $1 \le i \le BS-1$; and as a $Erlang\,(\mu_2, BS-1)$ R.V. w.p. ρ_2^{BS}. Thus we can derive the tail distribution of $\tilde{W}_2$:

$$P\left\{\tilde{W}_2 > t\right\} = \begin{cases} 0 \text{ if } BS=1 \\ e^{-\mu_2 t}\sum_{k=0}^{BS-2}\frac{(\mu_2 t)^k}{k!}\rho_2^{k+1} \text{ if } BS \ge 2 \end{cases}. \tag{2.23}$$

Using (2.22) and (2.23), we obtain (2.14).

Proof of Lemma 1

There are three possible events: Arrival, Completion 1, and Completion 2. Note that only Completion 1 increases $\delta(q_1, q_2)$, while the other two events decrease $\delta(q_1, q_2)$. Specifically, after any Completion 1 the system's state changes to $(q_1-1, q_2+1, s-1)$, so that $\delta(q_1-1, q_2+1) =$

$\delta(q_1, q_2) + 2$. In the worst case, all the events until stoppage are Completion 1 events. Then, from state (q_1, q_2, s), $R^{q_1,q_2,s}$ times consecutive Completion 1 events would lead to a stoppage at Station 1, where $R^{q_1,q_2,s}$ is given by

$$R^{q_1,q_2,s} = \begin{cases} \frac{TH-\delta(q_1,q_2)}{2} & if\ TH - \delta(q_1, q_2)\ is\ even, \\ \frac{TH+1-\delta(q_1,q_2)}{2} & if\ TH - \delta(q_1, q_2)\ is\ odd. \end{cases}$$

Specially if $\delta \leq TH - 2s + 1$, then $s \leq R^{q_1,q_2,s}$ and the TC gets into Station 2 before any stoppage could happen. There will be no stoppage for the TC.

Otherwise, if $\delta > TH - 2s + 1$ the TC may be stopped once (or more). In the worst case, after Station 1 starts working again the system experiences a repeating set of sequential events: Completion 1⇒Arrival(or Completion 2)⇒Arrival(or Completion 2). Each set of sequential events has two stoppages. So we get

$$\begin{aligned} M^{q_1,q_2,s} &= \begin{cases} 2(s - \frac{TH-\delta(q_1,q_2)}{2} - 1) + 1 & if\ TH - \delta(q_1, q_2)\ is\ even, \\ 2(s - \frac{TH+1-\delta(q_1,q_2)}{2} - 1) + 2 & if\ TH - \delta(q_1, q_2)\ is\ odd, \end{cases} \\ &= 2s - TH + \delta(q_1, q_2) - 1. \end{aligned}$$

Proof of Proposition 4

(1) If $0 \leq n < q_2$, the random walk cannot visit any points on the line $x = y$ where starvation occurs; thus $P\{B_2^{q_1,q_2,s} = 0\} = 1$. The sample path with solid line in Figure 2.5 is an example. The number of paths from $(q_2, 0)$ to $(q_2 + s - 1, n)$ is $\binom{n+s-1}{n}$. In any one of these paths, $s - 1$ moves should be to the right and n moves should be up, so each path occurs with probability $\left(\frac{\mu_1}{\mu_1+\mu_2}\right)^{s-1}\left(\frac{\mu_2}{\mu_1+\mu_2}\right)^{n}$. Thus, the probability that the random walk starting from $(q_2, 0)$ ends in $(q_2 + s - 1, n)$ is $Binomial\left(n; n + s - 1, \frac{\mu_2}{\mu_1+\mu_2}\right) = \binom{n+s-1}{n}\left(\frac{\mu_1}{\mu_1+\mu_2}\right)^{s-1}\left(\frac{\mu_2}{\mu_1+\mu_2}\right)^{n}$. The last move must always be to the right, so that for $0 \leq n < q_2$,

$$\begin{aligned} & P\{q_2 + s - 1 - K^{q_1,q_2,s} = n, B_2^{q_1,q_2,s} = 0\} = P\{q_2 + s - 1 - K^{q_1,q_2,s} = n\} \\ = & \binom{n+s-1}{n}\left(\frac{\mu_1}{\mu_1+\mu_2}\right)^{s}\left(\frac{\mu_2}{\mu_1+\mu_2}\right)^{n}. \end{aligned} \tag{2.24}$$

Eq. (2.24) corresponds to a Negative-Binomial distribution with parameters n, $n+s-1$ and $\frac{\mu_2}{\mu_1+\mu_2}$.

When the TC enters Station 2, she will then see $k = q_2 + s - 1 - n$ customers there, so from (2.24), $P\{K^{q_1,q_2,s} = k\}$ for $s-1 < k \leq q_2 + s - 1$, can be written as in the corresponding expression in (2.18).

(2) If $q_2 \leq n < q_2 + s - 1$, the RW can visit some points with starvation. The sample path with dashed line in Figure 2.5 is an example. Assume the number of points with starvation on this random walk is $B_2^{q_1,q_2,s} = i$. Then, the number of points with no starvation on this random walk is $s + n - 1 - i$. Each path occurs with probability $\left(\frac{\mu_1}{\mu_1+\mu_2}\right)^{s-i}\left(\frac{\mu_2}{\mu_1+\mu_2}\right)^{n}$.

Next, we calculate the number of lattice paths connecting $(q_2, 0)$ and $(q_2 + s - 1, n)$ that do not cross the line $x = y$ but have exactly i points in common with it. This number equals the number of lattice paths connecting the origin and $(s-1, n)$ that do not cross the line $y = x + q_2$ and have exactly i points in common with it and can be calculated by applying Corollary 1 in Milch (1970):

$$\begin{cases} \binom{s+n-1}{s-1} - \binom{s+n-1}{s+q_2-1} \; if \; i = 0, \\ \binom{s+n-i-1}{s+q_2-2} - \binom{s+n-i-1}{s+q_2-1} \; if \; i > 0. \end{cases} \tag{2.25}$$

Thus, the probability that a path starts from $(q_2, 0)$ and ends at $(q_2 + s - 1, n)$ is:

$$\begin{aligned} & P\{q_2 + s - 1 - K^{q_1,q_2,s} = n, B_2^{q_1,q_2,s} = i\} \\ = & \begin{cases} \left[\binom{s+n-1}{s-1} - \binom{s+n-1}{s+q_2-1}\right]\left(\frac{\mu_1}{\mu_1+\mu_2}\right)^{s}\left(\frac{\mu_2}{\mu_1+\mu_2}\right)^{n} & i = 0, \\ \left[\binom{s+n-i-1}{s+q_2-2} - \binom{s+n-i-1}{s+q_2-1}\right]\left(\frac{\mu_1}{\mu_1+\mu_2}\right)^{s-i}\left(\frac{\mu_2}{\mu_1+\mu_2}\right)^{n} & 1 \leq i \leq n - q_2 + 1. \end{cases} \end{aligned} \tag{2.26}$$

Because $P\{K^{q_1,q_2,s} = k\} = \sum_{i=0}^{s-k} P\left\{q_2 + s - 1 - K^{q_1,q_2,s} = q_2 + s - 1 - k, B_2^{q_1,q_2,s} = i\right\}$, using (2.26), the corresponding expression in $P\{K^{q_1,q_2,s} = k\}$, for $0 < k \leq s - 1$, in (2.18) follows.

(3) If $n = q_2 + s - 1$, the only path to the point $(q_2 + s, q_2 + s - 1)$ from the line $x = q_2 + s - 1$ is via $(q_2 + s - 1, q_2 + s - 2)$, to $(q_2 + s - 1, q_2 + s - 1)$ and then to $(q_2 + s, q_2 + s - 1)$. This can be seen as the sample path with pointed line in Figure 2.5. So the number of possible paths from $(q_2, 0)$ to $(q_2 + s - 1, q_2 + s - 1)$ is the same as the number of paths to $(q_2 + s - 1, q_2 + s - 2)$,

which can be calculated from (2.25) for $n = q_2 + s - 2$:

$$\begin{cases} \binom{2s+q_2-3}{s-1} - \binom{2s+q_2-3}{s+q_2-1} & i^* = 0, \\ \binom{2s+q_2-i^*-3}{s+q_2-2} - \binom{2s+q_2-i^*-3}{s+q_2-1} & 0 < i^* \leq n - q_2, \end{cases}$$

where, i^* is the number of times the random walk touches the line $x = y$ until it gets to $(q_2 + s - 1, q_2 + s - 2)$. Then, the number of times the random walk touches the line $x = y$ until it gets to $(q_2 + s, q_2 + s - 1)$ is $i = i^* - 1$; so

$$\begin{aligned} & P\{q_2 + s - 1 - K^{q_1,q_2,s} = n, B_2^{q_1,q_2,s} = i\} \\ = & \begin{cases} \left[\binom{s+n-2}{s-1} - \binom{s+n-2}{s+q_2-1}\right] \left(\frac{\mu_1}{\mu_1+\mu_2}\right)^{s-1} \left(\frac{\mu_2}{\mu_1+\mu_2}\right)^n & i = 1, \\ \left[\binom{s+n-i-1}{s+q_2-2} - \binom{s+n-i-1}{s+q_2-1}\right] \left(\frac{\mu_1}{\mu_1+\mu_2}\right)^{s-i} \left(\frac{\mu_2}{\mu_1+\mu_2}\right)^n & 2 \leq i \leq n - q_2 + 1. \end{cases} \end{aligned} \tag{2.27}$$

Because in this case when $n = q_2 + s - 1$ the TC sees no customer when she enters Station 2, we have $P\{K^{q_1,q_2,s} = 0\} = \sum_{i=1}^{s} P\{q_2 + s - 1 - K^{q_1,q_2,s} = q_2 + s - 1, B_2^{q_1,q_2,s} = i\}$, from which, we get the corresponding expression in $P\{K^{q_1,q_2,s} = k\}$, for $k = 0$, in (2.18).
It can be verified that $\sum_{i=0}^{q_2+s-1} P\{K = i\} = 1$.

Proof of Proposition 5

Using the Law of Total Probability:

$$P\{B_2^{q_1,q_2,s} = i\} = \sum_{n=0}^{q_2+s-1} P\{q_2 + s - 1 - K^{q_1,q_2,s} = n, B_2^{q_1,q_2,s} = i\} \qquad i = 0, 1, ..., s,$$

we can get $P\{B_2^{q_1,q_2,s} = i\}$ by conditioning on the value of i: When $i = 0$, using (2.24) and (2.26) , we get the corresponding expression in (2.20). When $i = 1$ and $i \geq 2$, using (2.26) and (2.27) , we get the corresponding expression in (2.20).

2.9.2 Algorithms

Algorithm 1 *: Calculate the distribution of $B_1^{s,\delta}$, the number of stoppages experienced by a TC from a state (s, δ) for each $s = 1, 2, \ldots, Limit$ and $\delta = -Limit, -Limit + 1 \ldots, TH + 1$.*

Step 1*: Let $s = 1$, $\delta = -Limit$.*

Step 2: *If* $\delta \leq TH - 2s + 1$, *set* $P\left\{B_1^{s,\delta} = 0\right\} = 1$.

Step 3: *If* $\delta = TH$ *or* $TH + 1$, *set* $P\left\{B_1^{s,\delta} = i\right\} = P\left\{B_1^{s,\delta-1} = i - 1\right\}$, $i = 1, ..., M^{s,\delta}$.

Step 4: *If* $TH - 2s + 1 < \delta \leq TH - 1$, *set* $P\left\{B_1^{s,\delta} = i\right\} = \frac{\lambda+\mu_2}{\lambda+\mu_1+\mu_2} P\left\{B_1^{s,\delta-1} = i\right\} + \frac{\mu_1}{\lambda+\mu_1+\mu_2} P\left\{B_1^{s-1,\delta+2} = i\right\}$, $i = 0, ..., M^{s,\delta}$.

Step 5: *Let* $\delta = \delta + 1$. *If* $\delta \leq TH + 1$, *then go to* ***Step 2****; else let* $s = s + 1$. *If* $s \leq Limit$, *then let* $\delta = -Limit$ *and go to* ***Step 2****. Otherwise,* ***Stop****.*

Algorithm 2 *: Calculate the distribution of* $K^{q_1,q_2,s}$, *the number of customers the TC sees when she enters Station 2 for* $s = 1, 2, \ldots, Limit$, $q_1 = s, s+1, \ldots, Limit$ *and* $q_2 = 0, 1, \ldots, Limit$.

Step 1: *Let* $s = 1$, $q_1 = Limit$, $q_2 = 0$.

Step 2: *If* $\delta \leq TH - 2s + 1$, *calculate the distribution of* $K^{q_1,q_2,s}$ *according to* (2.18).

Step 3: *If* $\delta = TH$ *or* $TH+1$, *set* $P\{K^{q_1,q_2,s} = i\} = \frac{\lambda}{\lambda+\mu_2} P\{K^{q_1+1,q_2,s} = i\} + \frac{\mu_2}{\lambda+\mu_2} P\{K^{q_1,q_2-1,s} = i\}$, $i \in [0, q_2 + s - 1]$.

Step 4: *If* $TH-2s+1 < \delta \leq TH-1$ *and* $q_2 = 0$, *set* $P\left\{K^{q_1,0,s} = i\right\} = \frac{\lambda}{\lambda+\mu_1} P\left\{K^{q_1+1,0,s} = i\right\} + \frac{\mu_1}{\lambda+\mu_1} P\left\{K^{q_1-1,1,s-1} = i\right\}$, $i \in [0, s - 1]$.

Step 5: *If* $TH-2s+1 < \delta \leq TH-1$ *and* $q_2 \neq 0$, *set* $P\{K^{q_1,q_2,s} = i\} = \frac{\lambda}{\lambda+\mu_1+\mu_2} P\{K^{q_1+1,q2,s} = i\} + \frac{\mu_1}{\lambda+\mu_1+\mu_2} P\{K^{q_1-1,q_2+1,s-1} = i\} + \frac{\mu_2}{\lambda+\mu_1+\mu_2} P\{K^{q_1,q_2-1,s} = i\}$ $i \in [0, q_2 + s - 1]$.

Step 6: *Let* $q_2 = q_2 + 1$. *If* $q_2 \leq Limit$, *then go to* ***Step 2****; else let* $q_1 = q_1 - 1$. *If* $q_1 \geq s$, *then let* $q_2 = 0$ *and go to* ***Step 2****; else let* $s = s + 1$. *If* $s \leq Limit$, *then let* $q_1 = Limit$, $q_2 = 0$ *and go to* ***Step 2****; else* ***Stop****.*

Algorithm 3 *: Calculate the distribution of* $B_2^{q_1,q_2,s}$, *the number of times Station 2 waits since the TC arrives to the network and until she finishes service at Station 1 for* $s = 1, 2, \ldots, Limit$, $q_1 = s, s + 1, \ldots, Limit$ *and* $q_2 = 0, 1, \ldots, Limit$.

Step 1: *Let* $s = 1$, $q_1 = Limit$, $q_2 = 0$.

Step 2: *If* $\delta \leq TH - 2s + 1$, *calculate the distribution of* $B_2^{q_1,q_2,s}$ *using (2.20).*

Step 3: *If* $\delta = TH$ *or* $TH+1$, *set* $P\left\{B_2^{q_1,q_2,s} = i\right\} = \frac{\lambda}{\lambda+\mu_2} P\left\{B_2^{q_1+1,q_2,s} = i\right\} + \frac{\mu_2}{\lambda+\mu_2} P\left\{B_2^{q_1,q_2-1,s} = i\right\}$, $i \in [0, 1, \ldots, s]$.

Step 4: *If* $TH-2s+1 < \delta \leq TH-1$ *and* $q_2 = 0$ *set* $P\left\{B_2^{q_1,0,s} = i\right\} = \frac{\lambda}{\lambda+\mu_1} P\left\{B_2^{q_1+1,0,s} = i\right\} + \frac{\mu_1}{\lambda+\mu_1} P\left\{B_2^{q_1-1,1,s-1} = i - 1\right\}$, $i \in [1, 2, \ldots, s]$.

Step 5: *If* $TH-2s+1 < \delta \leq TH-1$ *and* $q_2 \neq 0$, *set* $P\left\{B_2^{q_1,q_2,s} = i\right\} = \frac{\lambda}{\lambda+\mu_1+\mu_2} P\left\{B_2^{q_1+1,q_2,s} = i\right\} +$

$\frac{\mu_1}{\lambda+\mu_1+\mu_2} P\left\{B_2^{q_1-1,q_2+1,s-1}=i\right\} + \frac{\mu_2}{\lambda+\mu_1+\mu_2} P\left\{B_2^{q_1,q_2-1,s}=i\right\}$, $i \in [0,1,\ldots,s]$.

Step 6*: Let $q_2 = q_2+1$. If $q_2 \leq Limit$, then go to **Step 2**; else let $q_1 = q_1-1$. If $q_1 \geq s$, then let $q_2 = 0$ and go to **Step 2**; else let $s = s+1$. If $s \leq Limit$, then let $q_1 = Limit$, $q_2 = 0$ and go to **Step 2**; else **Stop**.*

2.9.3 Example: Tandem Queue Network with Three Stations

We define the following TBP: $\delta_1 = q_2 - q_1,\ \delta_2 = q_3 - q_2$. Thus for given values of thresholds TH_i, station i is idled if $q_{i+1} - q_i \geq TH_i,\ i = 1,2$. To test the performance of this TBP we constructed a simulation model for a system with $\lambda = .85, \mu_1 = 1, \mu_2 = .95, \mu_3 = .9$. Station 3 is the bottleneck and the system utilization is $\rho = \rho_3 = .85/.9 = .94$. Note that this simply inserts an intermediate station into the network with $\rho = .94$ analyzed on Figure 2.7 in Section 2.6.1.

We simulated $500,000$ customers under the non-idling policy and the TBP with all possible combinations of TH_1 and TH_2, ranging from 5 to 100. For $t = 32$, we plot the lower envelope of the performance measures $\left(E\left[S^{TBP}\right], PW^{TBP}(32)\right)$ in Figure 2.10 as the solid curve. We observe that it has the similar behavior as the trade-off curves in Figure 2.7. The TBP with $TH_1 = 10$ and $TH_2 = 14$ achieves the maximum improvement of 42% in $PW(t)$ (6.43% vs 3.73%) at the cost of increasing the $E[S]$ by about 5.73% (from 34.46 to 36.44). Note that for the non-idling policy, the theoretical values for both $E[S]$ and $PW^{NI}(t)$ can be calculated. These values closely matched the values observed in our simulation, which validated the simulation model. Most of the observations made for the 2-station system earlier appear to apply to the 3-station system as well: the improvement in $PW(t)$ achieved by the TBP strongly depends on the value of t. Below certain t (the critical value is around 10 in this example), the TBP cannot improve over the NI policy at all. Above this critical value, the level of improvement grows with t.

Next, we compare the performance of Kanban and TBP for this system. The Kanban policy is defined by the values of buffer sizes BS_1 and BS_2; station $i-1$ is idled if $q_i \geq BS_{i-1}$, $i = 2,3$. We simulated the system for all combinations of values of $BS_1, BS_2 \in \{1,\ldots,100\}$. The lower envelope of the performance measures $\left(E\left[S^{Kanban}\right], PW^{Kanban}(32)\right)$ for the Kanban policy is

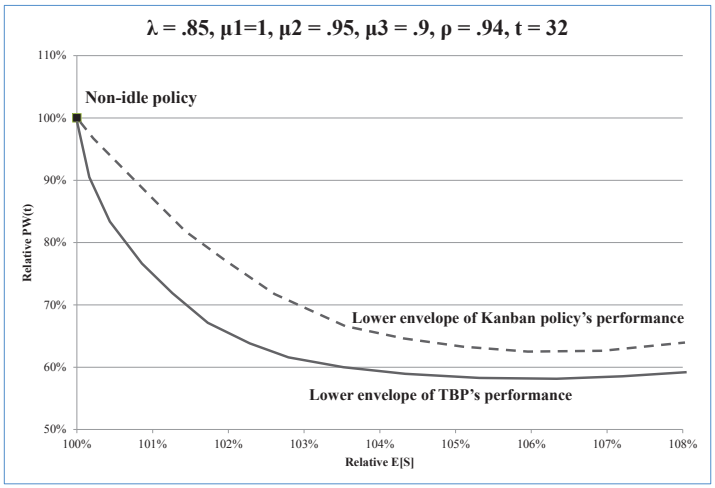

Figure 2.10: The lower envelope of the TBP's and Kanban policy's performances.

plotted in Figure 2.10 as the dashed line. The largest reduction in $PW(t)$ under the Kanban policy is 37.69% and occurs for buffer sizes $(BS_1, BS_2) = (24, 27)$.

We note that in this example, the TBP dominates the Kanban policy: for every feasible value of $E[S]$ the TBP achieves greater reduction in $PW(t)$ than the (optimized) Kanban policy. This performance was also observed in a number of runs with different parameter settings. We also observed that the TBP appears to be more robust: while the TBP with $(TH_1, TH_2) = (10, 14)$ performs well for a wide range of values of t, the optimal buffer sizes under the Kanban policies are quite sensitive to t. More detailed results are available upon request.

2.9.4 Generalization of TBP to n-station Tandem Queue Systems

Next, we give a generalization of TBP in a n-station tandem queue system. Consider a serial queueing network consisting of n stations, and let $PW(t) = \frac{1}{n}\sum_n P(W_i > t)$. To define a general TBP we specify for each station $j = 1, \ldots, n-1$ a function $\delta_j(q_j, \ldots, q_n)$ and $\delta_j(0, \ldots, 0) = 0$. (It is sensible to choose a function that is decreasing in q_j and increasing in q_k for all $n \geq k \geq j$. Then, when q_k is large or q_j is small, Station j is idled, and when q_k is small or q_j is large, Station j resumes working.) We also specify a threshold $TH_j \geq 0$ for all $j = 1, \ldots, n-1$. The TBP is now defined as follows: station j is idled whenever $TH_j \leq \delta_j(q_j, \ldots, q_n),\ j = 1, \ldots, n-1$. In the previous two sections we used $\delta_1(q_1, q_2) = q_2 - q_1$ that clearly satisfies the definition above. The general idea of the TBP remains the same: idle

an upstream station when the queues downstream are too long. However the definition above allows for a great deal of flexibility: the function δ_i could be more heavily weighted towards the bottleneck stations, taking into consideration all downstream queues, or only the immediate successor to the current station, normalize the queue at each station by the expected service time and or the number of servers at this station, etc.

2.9.5 Notation

π_{q_1,q_2}	The steady state probability of the $MC(Q_1, Q_2)$.
$\hat{M}^{q_1,q_2}$	The number of stoppages the TC will see, given she arrives at state (q_1, q_2).
$\hat{W}_i^{q_1,q_2}$	The TC's waiting time for Station i $(i = 1, 2)$, given she arrives at state (q_1, q_2).
$L_X(h)$	Laplace Transform of a R.V. X.
$X^{q_1,q_2,s}$	The TC's performance measure in the future, given the TCMC is now in (q_1, q_2, s).
$M^{q_1,q_2,s}$	The maximum number of stoppages the tagged customer will see, given the TCMC is in (q_1, q_2, s).
$R^{q_1,q_2,s}$	Number of sequential times Completion 1 needs to occur, until Station 1 would be idled given the TCMC is in (q_1, q_2, s).
$W_1^{s,\delta}$	The TC's waiting time for Station 1, given the revised TCMC is in (s, δ).
$B_1^{s,\delta}$	Number of stoppages in Station 1 the TC may experience, given the revised TCMC is in (s, δ).
$W_2^{q_1,q_2,s}$	The TC's waiting time for Station 2, given the TCMC is in (q_1, q_2, s).
$K^{q_1,q_2,s}$	Number of customers the TC may see when she enters queue 2, given the TCMC is in (q_1, q_2, s).
$B_2^{q_1,q_2,s}$	Number of times that Station 2 is starved, given the TCMC is in (q_1, q_2, s).
$S^{q_1,q_2,s}$	The TC's sojourn time, given the TCMC is in (q_1, q_2, s).

Table 2.2: Notation

Chapter 3

Dynamic Scheduling and Strategic Idling in an Open-shop Service Network: Case Study and Analysis

3.1 Introduction

The primary goal of this paper is to develop effective dynamic scheduling policies for stochastic open-shop processes operating under multiple objectives. The system objectives may include a combination of the more traditional "macro-level" measures (such as minimizing total system time and minimizing total tardiness) with the "micro-level" objective seeking to limit the number of incidents where a customer experiences an "excessively long" wait at a workstation within the process. This combination of objectives is motivated by the understanding that customer's perception of service quality is affected by both "macro" and "micro" level factors.

Open-shop service networks, where customers need to visit a set of stations without a specific service order (other than some possible precedence constraints), are quite common in modern service industry in both, the "brick and mortar" and "virtual service" operations. Examples of the former include retail stores, where customers may visit several departments before proceeding to the cashier, and hospitals, where patients often go through several diagnostic and treatment stations. Examples of virtual systems include contact centers, where calls may

be served by a mixture of automated response units and human agents with different levels of expertise, and on-line websites, where customers typically visit a number of pages before proceeding to the checkout page.

This paper was motivated by the example of an open shop in the healthcare service industry operated by XYZ (the real name of the company is removed and relevant data has been disguised for confidentially reasons) – one of the leaders in preventive healthcare services in North America. Their flagship service is composed of 10-20 different medical tests, each test performed in a different station. This service provides customers with a comprehensive evaluation of their current state of health and allows them to actively manage their health care. The order in which customers take most of these tests is immaterial (there are only a few precedence constraints), so XYZ actually operates an open-shop service system. XYZ's service is primarily targeted towards busy professionals who are willing, for a fee, to have a complete assessment of their health performed in just a few hours. It should be noted that under the Canadian medical system, most of the tests can be done for free, but would likely take days or even months to schedule and complete. Thus, convenience is the main selling feature, and delivering excellent customer service is of paramount importance for XYZ. Tight management of customer waiting time in the system is deemed to be essential. While, due to the inherent variability of the times it takes to perform the different tests and procedures, some waiting time is unavoidable, the goal is to minimize waiting times and maximize customer perception of service quality. To study the waiting times in XYZ we interviewed their key personal and collected two months of data, comprising 41 business days with just over 2000 customers and about 24,000 station visits.

While there are many determinants of service quality in service networks, the link between customer waiting times and the perceived service quality is well-recognized (Friedman and Friedman, 1997; Taylor, 1994). Waiting times, that are easily quantifiable, have long been the focus of much of the queueing literature. The most common measure of waiting time is the expected overall waiting time for service. A related measure is the probability that the total system time or the total waiting time exceeds a certain pre-determined threshold. XYZ uses both of these service level measures (SLMs) with the following stated targets: mean system time of less than four hours, and the probability of system time longer than four hours of less

than 50%.

The two SLMs described above take a "macro" view of the service network, essentially treating it as a one-stage system and looking at the overall system or waiting time. Such SLMs may be sufficient in the manufacturing context, where customers, after placing an order, remain "outside the system" and essentially view the system as a "black box". However, in service contexts the customer is not just an outside observer: they experience the internal performance of the system as well, i.e., waiting times in front of individual workstations. A poor experience at a given workstation may lead to a poor perceived service quality, even when the macro-level SLM (e.g., the overall waiting time) does not indicate a problem.

In fact, XYZ's senior management noticed that there have been two types of complaints about the service experience: the first one is with respect to the total time customers spend in the system and the second, more prevalent one, is with respect to a long wait for a particular station. Specifically, from customer satisfaction surveys XYZ found that customers whose wait for service at any station exceeded 20 minutes were substantially less satisfied with their service than others. This led XYZ to define a "micro-level" SLM: the waiting time in front of any station should not exceed 20 minutes. This measure is taken very seriously: when a customer wait time at a station reaches 15 minutes an anxious "yellow face" appears on process schedulers' screens, alerting them to a potential issue. Once the waiting time reaches 20 minutes, an angry "red face" appears, and a service breakdown is considered to have occurred. The total number of red faces is carefully tracked: the goal is to keep this number below 100 per month.

The importance of such micro-level measures has been observed in other settings. For example, Bouch, Kuchinsky and Bhatti (2000) show that for an on-line service, when a single web page takes longer than 8 seconds to load, customer service quality ratings fall off dramatically. The adverse impact of long waiting time at a particular station is further supported by the marketing literature, e.g., Soman and Shi (2003) show that, given the total system time and price, people prefer a situation in which they are constantly making progress towards their goal; and by the psychology of queueing literature, e.g., Larson (1987) shows that customers' perception of the queueing experience may vary nonlinearly with the delay.

Nevertheless, the systematic treatment of such micro-level SLMs in queuing literature is relatively new. The only prior paper we are aware of is Baron, et al. (2013), where it was

demonstrated analytically for a two station tandem queue network, that a scheduling policy with *Strategic Idling* (SI) might be helpful in reducing the probability of long waits. The idea behind SI policies is that when a downstream station accumulates a long queue, continuing to operate upstream stations at the normal rate may lengthen the queues downstream and increase the probability of long waits (i.e., the expected number of red faces). A better idea may be to idle the upstream stations (or to temporarily reduce their service rate), allowing the downstream queues to dissipate. A wise use of such idling effectively re-distributes the waiting times more evenly among the stations and reduces the number of red faces. We note that the scheduling policies employing SI violate one of the more common assumptions in queuing analysis: the "work conserving" property, which states that a workstation should continue to operate as long as there are customers waiting to be served. However, the potential payoff of SI may be very attractive. Indeed the "classical" way of reducing probabilities of long waits is by adding capacity to the system (e.g., adding a doctor in the healthcare setting), which is often quite expensive. However, using a dynamic scheduling policy with SI can potentially achieve the same objective at a negligible (or even a negative) cost: by simply idling some resources in the system.

Given the results from Baron et al. (2013) discussed above, and the fact that the number of red faces is used as part of performance evaluation for process schedulers at XYZ, one may expect that the schedulers will use the SI strategy. While our interviews with XYZ's management team indicate that such use of SI is not company's policy, our analysis of the empirical data points to convincing statistical evidence that XYZ's schedulers in fact do employ SI quite effectively to manage the number of red faces - simulation results indicate that the current scheduling rules without the use of SI would likely result in more than twice the current number of red face incidents.

Our primary interest is to study the benefit of dynamic scheduling policies (DSPs) and SI for open-shop service networks, such as the one operated by XYZ. We start by developing a framework that allows us to represent and implement general dynamic scheduling policies as simple scoring rules. A variety of different DSPs based on intuition and known theoretical results for stylized systems are proposed to address the macro-level SLMs. We then show how a *threshold-based policy* approach can be used to intelligently inject SI into a given DSP, resulting

in a policy that can potentially address both macro and micro-level objectives simultaneously.

To test our policies we develop a simulation model for the XYZ process. The need for the simulation-based approach is driven by both, the complexity of stochastic open-shop networks, making analytical results very hard to obtain, and by the transient nature of real-life systems, such as the one operated by XYZ: since the system starts each workday with an empty queue and ends it in the same state after seeing relatively few customers, the steady-state regime may never set in. We use the empirical data on arrivals and service times to first understand the service process at XYZ and then to calibrate a detailed simulation model. Using this model we investigate the performance of several DSPs with and without SI and compare them to the performance of the empirical scheduling policies used in XYZ. We show that the automated policies achieve very promising results: the best DSPs are able to significantly outperform the actual schedules with respect to the macro-level measures. While without the use of SI the DSPs tend to perform poorly on the micro-level measures (the same effect is demonstrated for the actual scheduling policies), after the SI modification, DSPs are able to perform very competitively on the micro-level SLM, while maintaining their advantage with respect to macro-level SLMs.

To summarize, the paper makes several key contributions: (1) we narrow the gap between theory and practice on scheduling in open-shop service networks; (2) we shed light on the need for effective and systematic implementation of DSPs and SI to improve the SLM in such networks; (3) we provide a framework for developing algorithms for the joint usage of DSPs and SI in open-shop service systems; (4) using such algorithms and the simulation model of XYZ we demonstrate that an open-shop service network can be managed in a *systematic fashion* to deliver improved service level by jointly using DSPs and SI; and (5) we establish the usage of SI in practice (using statistical tests).

The plan for the remainder of the paper is as follows. In Section 3.2, we provide a brief literature review, focusing on known results for open-shop systems. In Section 3.3, we introduce the frameworks for using DSPs and SI in a stochastic open-shop service network. Then, using these frameworks, we propose several stylized DSPs and SI policies that are likely to be successful in practice. In Section 3.4, we analyze the empirical data provided by XYZ and present evidence for the usage of SI in XYZ. In Section 3.5, we use a simulation model to demonstrate the effect

of SI and evaluate some of the DSPs and SI policies suggested in Section 3.3. In Section 3.6, we present conclusions and suggestions for future research.

3.2 Literature Review

Open shops were studied extensively in manufacturing settings (see, e.g., Roemer 2006), such as airplane maintenance, fire engine assembly, just-in-time systems, supply chain assembly systems, paper processing, and part kitting. A few papers consider service systems, such as accounting services, but they still consider common manufacturing measures as the service objective.

Open-shop problems are typically NP-hard and only consider macro-level measures (i.e., see Pinedo 2012 and reference therein). For the deterministic open shop with preemptions, polynomial time algorithms are available for the makespan objective, and maximum lateness objective. Without preemptions, for the makespan objective, only open shop with $m = 2$ stations has the polynomial-time optimal policy (the Longest Alternate Processing Time first policy), and open shop with $m \geq 3$ stations is known to be NP-hard. For the maximum lateness objective, open shop with $m = 2$ stations is already strongly NP-hard. Furthermore, very little can be said about the total completion time objective $\sum_{i=1}^{n} F_i$; open shops with this objective is NP-hard for all $m \geq 2$ cases, with or without preemptions.

For the stochastic open shops, theoretical results are limited to the $m = 2$ case. Pinedo and Ross (1982) proved that the Longest Expected Remaining Processing time first (LERP) policy minimizes the expected makespan of a stochastic two-station open shop. Pinedo (1984) showed that the preemptive Shortest Expected Remaining Processing time first (SERP) policy minimizes the total expected completion time in a two-station open shop within the class of preemptive dynamic policies.

Alcaide et al (2006) developed a predictive-reactive approach to minimize expected makespan in an open shop with $m \geq 3$ stations; the approach is based on dynamically modifying a heuristic schedule, based on Alcaide et al. 1997, whenever an unexpected event occurs.

A vast literature is focused on the analysis, design, and control of queueing networks; see Stidham (2002) for a thorough survey of this research. Another important stream of queueing

literature is focused on scheduling policies, which assign priorities to customers (or jobs) based on the current state of the system and the attributes of all customers. Scheduling policies can be categorized into two classes: static and dynamic. Static scheduling policies assign priority to each customer by a static rule which does not change while the customer is in the system. For example, to minimize customers' average waiting cost in a system with linear waiting cost rates, the $c\mu$ rule (see, e.g., Smith 1956) assigns static priority levels to customer k in an increasing order of $c_k\mu_k$ where c is the cost of waiting and μ^{-1} is the expected service time. Dynamic scheduling policies, in contrast, assign priorities that may be changed while customers are in the system. For example, to minimize customers' waiting cost in a system with convex waiting cost, the generalized $c\mu$ rule (see, e.g., Van Mieghem 1995) assigns dynamic priority levels to each customer k according to $c_k'\left(W_k\left(t\right)\right)\mu_k\left(t\right)$, where $W_k\left(t\right)$ is customer k's waiting time at time t, and $c_k'\left(\cdot\right)$ is the first derivative of $c_k\left(\cdot\right)$.

Harrison (1996) used fluid models to provide asymptotically optimal scheduling heuristics under different objectives. This approach was extended by Maglaras (2000) who proposed discrete-review policies to translate the solution of the fluid optimal control into an implementable control policy in the stochastic network. For the job-shop scheduling problem with holding cost objective, Bertsimas et al. (2003) provided an efficient algorithm to round an optimal fluid solution such that the resulting schedule is asymptotically optimal. Dai and Lin (2005) proved that maximum pressure policies are throughput optimal in a class of stochastic processing networks. We do not use asymptotic approaches and directly consider scheduling in the stochastic network.

By far, the most popular objective in the literature employs is the total system time (see e.g., the survey by Gans, Koole, and Mandelbaum, 2003). A related measure is the probability that the total system time or the total waiting time exceeds a certain pre-determined threshold. Baron, Berman, and Krass (2008), Baron and Milner (2009), de-Véricourt and Jennings (2011) and references therein also focused on the probability of long waiting time SLM.

Several other papers looked beyond the traditional measures. For example, de-Véricourt and Zhou (2005) analyzed a call-routing problem while considering both the call resolution probability and the average service time in the macro-level service level measure. Mehrotra et al. (2012) considered a similar problem with heterogeneous servers. Saghafian, Hopp, and

Van Oyen (2012) analyzed the service policy in Emergency Departments while considering the weighted average of the expected length of stay and the expected time to first treatment.

The systematic study of the micro-level SLM focusing on the instances of excessive waits originates with Baron, et al. (2013), who demonstrate the advantage of policies with SI by applying a *Threshold Based Policy* (TBP). The idea behind the TBP is to compare the difference between queue lengths at different stations and to idle some upstream stations if this difference is larger than a predetermined threshold. They demonstrate the exact potential advantage of applying TBP in a tandem queue system with two servers.

There are two other settings where intentionally idling a capacitated resource has been previously considered. Strategic delays were first discussed in the literature in Afèche (2013), who showed how such delays can allow a service provider to differentiate between customer types and thus improve the overall profit. The manufacturing process control literature also considers intentional idleness. The most prominent example is the Kanban manufacturing system where the total inventory between two stations is restricted to be lower than a threshold - for further details see Masin, Herrar, and Dar-el (2010) and references therein. In this case the motivation for idling is the need to control inventory and its cost without sacrificing too much capacity. In both cases, the motivation for intentional idling is significantly different from the current paper, which is motivated by improving the customer service experience. This difference leads to completely different analysis and implementation challenges.

3.3 Dynamic Scheduling Policies and Strategic Idleness Modification

We start by introducing stochastic open shop with precedence constraints in Section 3.3.1. We next develop a framework of completely reactive Dynamic Scheduling Policies (DSPs) that allow us to represent different DSPs in terms of simple scoring rules in Section 3.3.2. Using this framework, in Section 3.3.3, we propose several simple DSPs that have the potential to perform well in practice with regards to the macro-level Service Level Measures (SLMs). In Section 3.3.4, we show how any completely reactive DSP can be modified to "inject" Strategic Idleness (SI), allowing the policy to take into account our micro-level "red faces" SLM.

3.3.1 Stochastic Open Shop with Precedence Constraints

We consider a general stochastic open-shop problem with precedence constraints described as follows: a set of n customers $C = \{1, \ldots, n\}$, who arrive at (possibly random) release dates $r_1^c, \ldots, r_n^c$, wish to finish service within T^o time units after arrival (i.e., their due dates are $r_1^c + T^o, \ldots, r_n^c + T^o$). The customers need to obtain service from a set of m stations $S = \{1, \ldots, m\}$ that open at pre-scheduled release dates $r_1^s, \ldots, r_n^s$ and close when all customers have finished service. For simplicity, we assume that $r_i^c < r_{i+1}^c$, for $i = 1, \ldots, n-1$, and $r_j^s < r_{j+1}^s$, for $j = 1, \ldots, m-1$, i.e., this implies no batch arrivals or stations openings at the same time. Customer i requires service from some subset $S_i \subseteq S$ of stations, and she must visit every station in S_i exactly once. The order in which customer i receives services from stations in S_i is immaterial, as long as it satisfies precedence constraints $U_i = \{(h,k), \ldots | h, k, \ldots \in S_i\}$, where constraint (h,k) means that customer i must visit station h before becoming eligible for station k, i.e., station h is a precedent station of station k in U_i. For example, $U_i = \{(1,2), (1,3), ..., (1,m) | 1, 2, ..., m \in S_i\}$ means that customer i needs to visit station 1 before visiting any other stations. Note that, if $U_i = \emptyset$ for all i, the problem becomes a classic open-shop problem (see, e.g., Pinedo 2012). Customer i's service time at station j (for $j \in S_i$), X_{ij}, is a *continuous* random variable with distribution G_j. We assume that X_{ij} are independent and identically distributed for all j. The realization x_{ij} only becomes known upon service completion. We consider the problem *without* preemptions (i.e., customers are not allowed to leave the service at the current station before completion, nor is a station allowed to accept new customers before finishing the service of the current customer). The time customer i finishes service and exits the system is denoted by F_i, which is also a random variable.

We treat the scheduling problem as a multi-objective problem. Specifically, we consider two macro-level SLMs as our objectives: minimize **the expected total lateness**,

$$E\left[\sum_{i=1}^{n}\left(F_i - (r_i^c + T^o)\right)\right], \tag{3.1}$$

and minimize **the expected number of tardy customers**,

$$E\left[\sum_{i=1}^{n} 1\left(F_i > (r_i^c + T^o)\right)\right]. \tag{3.2}$$

Note that since r_i^c and T^o are independent of the scheduling policy, the expected total lateness objective is equivalent to the more typical expected total system time objective, $E\left[\sum_{i=1}^{n}\left(F_i - r_i^c\right)\right]$, or the expected total completion time objective, $E\left[\sum_{i=1}^{n} F_i\right]$.

In addition to the macro-level SLMs above, we consider a micro-level SLM as our third objective: minimize **the expected number of "red faces"** (i.e., the number of instances of unacceptably long waits),

$$E\left[\sum_{i,j\in S_i} 1\left(W_{ij} > T^s\right)\right], \tag{3.3}$$

where T^s is the threshold used to identify "red faces" and W_{ij} is the random variable denoting the time customer i spent in the waiting room before entering station j.

In the manufacturing literature, DSPs that consider unexpected real-time events (e.g., station breakdown, defective material, job cancelation, etc.) have been classified into three categories (see, e.g., Ouelhadj and Petrovic, 2009): (1) a completely reactive scheduling policy generates no firm schedule in advance and makes decisions locally in real-time; (2) a predictive-reactive scheduling policy develops a schedule first and revises it in response to real-time events following some scheduling/rescheduling methods; and (3) a robust pro-active scheduling policy follows a pre-set schedule that satisfies performance requirements predictively in a dynamic environment. Note that this taxonomy can also be applied in stochastic scheduling models with uncertainties caused by stochastic service times, as well as other of unexpected events.

Since, as we detailed in the literature review above, polynomial time algorithms for deriving the optimal scheduling policy in open shops are not available, it is hard to generate any firm schedule in advance. Therefore the advantage of predictive-reactive or robust pro-active DSPs is difficult to see for the open-shop service network we are interested in. Thus, we focus our investigation on completely reactive DSPs. We will initially consider only work-conserving DSP, i.e., a customer cannot be waiting for a station which is currently idle.

3.3.2 Work-conserving Dynamic Scheduling Policies in Open Shops

The completely reactive work-conserving DSPs in an open-shop environment take actions at three types of events: service completions, customer arrivals, and station openings. At these points, either a customer, or a station, or both, free up and must be "matched up" with the available customers/stations for the next stage of processing. In fact, by introducing dummy stations and customers, it suffices to only consider service completion events. We can imagine that the system starts with n dummy stations serving n customers with service completion times equal to customer arrival times $r_1^c, \ldots, r_n^c$, and m stations serving m dummy customers with service completion times equal to station opening times $r_1^s, \ldots, r_n^s$. Henceforward, we only consider decisions at service completions events.

Following a common simplifying assumption in queueing and stochastic scheduling literature, we assume that no two service completions happen at the same time, i.e., there exists an $\epsilon > 0$, such that the time interval between any two service completions is at least ϵ. This assumption fits XYZ and simplifies the discussion and rules below.

Consider a service completion involving customer i and station j occurring at some time t. We assume that at this time customer i enters the "waiting room" (either physical or virtual) and station j becomes idle. Let $\Omega_j \subseteq C$ be the set of *eligible* customers (i.e., satisfy all precedence constraints) that still require service from station j and who are in the waiting room at time t, and $\Psi_i \subseteq S_i$ be the set of idle stations at time t whose services are still required by customer i. Since any customer i only visits stations in S_i once, we have $j \notin \Psi_i$ and $i \notin \Omega_j$ at time t. Also, for any waiting customer h other than customer i, Ψ_h is either $\emptyset$ or equal to $\{j\}$, so we have $\Psi_i \cap \Psi_h = \emptyset$ (note that if $k \in \Psi_h$ for some $k \neq j$ then customer h was waiting while station k was idle, violating the work-conserving assumption). Similarly, for any idle station $k \neq j$, Ω_k is either $\emptyset$ or $\{i\}$, so we have $\Omega_j \cap \Omega_k = \emptyset$. This indicates that, at each service completion, the DSP needs to perform at most two assignments: assign a customer $h \in \Omega_j$ to station j (assuming $\Omega_j \neq \emptyset$) and assign a station $k \in \Psi_i$ to customer i (assuming $\Psi_i \neq \emptyset$); no other assignments are possible.

These observations allow us to represent a DSP with two scoring rules, where higher is better. We assign a score, $PT_h^c \geq 0$, to customers $h \in \Omega_j$, and a score, $PT_k^s \geq 0$, to stations

$k \in \Psi_i$. To make PT_i^c and PT_j^s general enough, we define them as arbitrary non-negative functions of the history of the system up to time t.

Definition 1 *Completely reactive and work-conserving $\mathcal{DSP}$ in stochastic open-shop networks is defined by scoring rules PT_i^c and PT_j^s as follows:*
Suppose customer i completes service on station j at time t:
1) For the station assignment, if $\Omega_j \neq \emptyset$ we assign customer $h^* = \arg\max_{h\in\Omega_j} PT_h^c$ to be the next customer of station j; let station j stay idle if $\Omega_j = \emptyset$.
2) For the customer assignment, if $\Psi_i \neq \emptyset$ we assign station $k^* = \arg\max_{k\in\Psi_i} PT_k^s$ to be the next station of customer i; let customer i join the waiting room if $\Psi_i = \emptyset$.
Once these two assignments are made, the service process continuous until the next service completion event; then similar assignments are taken.

Note that the assumption that no two service completions occur simultaneously ensures that the definition above is complete. If, instead, several service completions occur at once, we can no longer assume that customer i is the only eligible waiting customer for any station in Ψ_i, nor that station j is the only idle station needed by any customer in Ω_j; there may be several customers "competing" for the same station and several stations "competing" for the same customer. Therefore, in addition to scoring rules, one must provide tie-breaking rules in order to specify the DSP in this case.

From Definition 1, we see that any DSP is completely determined by the selections of PT_i^c and PT_j^s. We next discuss different choices of these scoring rules that result in different DSPs.

3.3.3 Dynamic Scheduling Policies

To describe a DSP (or, more precisely, the scoring rules defining the policy) formally, we introduce the following notation (we omit t for convenience):
S_i^F: the set of stations customer i has visited by time t;
S_i^U: the set of stations customer i still requires service from at time t;
(Note that $S_i^F(t) \cup S_i^U(t) = S_i$, $\forall t$.)
u_j: the number of customers who still need service from station j at time t, i.e., $u_j = \sum_{i=1}^n 1\left(j \in S_i^U\right)$;

w_i^{TS}: customer i's total system time until time t, i.e., $w_i^{TS} = t - r_i^c$;

w_i^{TW}: customer i's total waiting time since she entered the system until time t;

w_i: customer i's current waiting time (i.e., the time since the last service completion of customer i and until time t);

$\bar{s}_j$: the average service time of station j;

n_j: the number of servers at station j.

Using the definitions above, for any $k \in \Psi_i$, the quantity $\frac{u_k \bar{s}_k}{n_k}$ represents the *remaining average workload* of station k. Since stations with the higher remaining average workload at time t can be thought of as "bottlenecks" over the remainder of the process, it seems reasonable to perform customer assignment so as not to keep more valuable resources idle. Thus, all DSPs we consider employ the same station scoring rule: $PT_k^S = \frac{u_k \bar{s}_k}{n_k}$, assigning customer i to the station with the highest remaining workload.

Our DSPs do differ with respect to station assignment rules (i.e., deciding which customer should be assigned next to the freed-up station j).

1. *Longest System time first (LS) policy* assigns to station j the customer in Ω_j who has the longest system time among all waiting customers who still require service from this station, i.e., $PT_i^c = w_i^{TS}$.

   This rule is motivated by the idea that the customer who has already accumulated a long system time (because of waits or long processing times) is more likely to be tardy, and thus should be prioritized.

2. *Longest Mean Overage Processing time first (LMOP) policy* assigns to station j the waiting customer in Ω_j who has the longest mean overage service time, i.e., $PT_i^c = \frac{1}{|S_i^F|} \sum_{k \in S_i^F} (x_{ik} - \bar{s}_k)$.

   Similar to the LS policy, LMOP policy prioritizes the customer who experienced longer than usual service times (represented by a longer mean overage service time). This customer thus has a higher risk of being tardy.

3. *Longest Accumulated Waiting time first (LAW) policy* assigns the waiting customer in Ω_j who has accumulated the longest waiting time, i.e., $PT_i^c = w_i^{TW}$.

This policy is also motivated by prioritizing customers who have a higher risk of being tardy. By counting only waiting time we are giving preference to customers who have already been "victimized" by long waits.

4. *Longest Current Waiting time first (LCW) policy* assigns the waiting customer in Ω_j who has the longest current waiting time, i.e., $PT_i^c = w_i$.

   This policy follows the spirit of first-come-first-serve policy and prioritizes customers who enter the centralized waiting room earlier.

5. *Shortest Expected Remaining Processing time first (SERP) policy* assigns the waiting customer in Ω_j who has the shortest total expected remaining processing time, i.e., $PT_i^c = \left(\sum_{k \in S_i^U} \bar{s}_k\right)^{-1}$.

   This policy is motivated by the optimality of preemptive SERP policy in a two-station open shop with the total expected completion time objective (see, e.g., Pinedo 1984).

6. *Longest Expected Remaining Processing time first (LERP) policy* assigns the waiting customer in Ω_j who has the longest total expected remaining processing time, i.e., $PT_i^c = \sum_{k \in S_i^U} \bar{s}_k$.

   This policy is motivated by the optimality of LERP policy in a two-station open shop with the expected makespan objective (see, e.g., Pinedo and Ross 1982).

3.3.4 Strategic Idleness Modification - Generalized TBP

Recall that in addition to the two macro-level objectives (SLMs), the total lateness and the total number of tardy customers, we are also interested in the micro-level objective: the total number of "red faces" (incidents of long waits). Since such incidents often occur at bottleneck stations, one strategy to reduce their number is to intentionally delay service at stations that are upstream from the bottlenecks when there is already a long queue in front of the bottleneck station. We call such intentional delays "Strategic Idleness" (SI). The overall idea is to modify a given work-conserving DSP so that when the DSP assigns a free customer to a free station, instead of starting the service immediately, they both stay idle for a certain time period (which is determined by the SI policy and could be zero) prior to the service start. Note that with

this modification the station remains assigned to a customer during SI period; thus customer and station assignments can be implemented using the same rules that are used to specify the original work-conserving DSPs. Note that non-idle policy can be thought of as a special SI modification where the idling time is always zero.

There are many possible policy classes that may involve SI. Baron et al. (2013) introduced a specific family of *Threshold Based Policies (TBP)*. As mentioned earlier, the idea behind the TBP is to compare the difference between queue lengths at different stations and to idle some upstream stations if this difference is larger than a predetermined threshold.

While defining a TBP in the two station tandem queue setting is straightforward, as there is only one upstream and one downstream station, it is already more difficult for a n-station tandem queue. The difficulty grows further in an open-shop service network, where not every customer may need to go through every station, and each customer may take a unique path through the network. Thus the "upstream" and "downstream" stations may not be clearly defined. To extend the definition of the TBP to this setting, we proceed as follows. Recall that at each time period, u_k gives the number of customers still requiring service from station k. For station j and customer i we define δ_{ij} to be a function of $u_1, \ldots, u_m$ which is decreasing in u_j, non-increasing in u_k for all $k \in \{1, \ldots, m\},\ k \neq j$, and $\delta_{ij}(0, \ldots, 0) = 0$. To make δ_{ij} customer-specific, we allow it to depend on the set S_i^U. For every station j, we also define a non-negative threshold TH_j.

Definition 2 *Modified policy DSP+TBP*
When customer i is ready to enter station j under DSP, delay the service starting time as long as $\delta_{ij}(u_1, \ldots, u_m) \geq TH_j$.

We say that customer i is *stopped* (at station j) if this customer is assigned to station j while this station is *idled*. Intuitively, the TBP modification will idle station j if the number of customers who still require this station is low compared to other stations in the system. Several examples are presented below.

When customer i is stopped at station j under the TBP, there are, in principle, two options: (1) we can serve the next customer waiting for station j, allowing this customer to overtake

customer i (provided this customer is not stopped under the TBP) - we call this "TBP with overtaking"[1]; (2) we can simply idle station j until the block is released under the TBP - leading to "Overtake-free TBP" (note that the latter option is more in line with the goal of minimizing excessive waits for the current customer).

The definition of TBP above is very flexible. The following specification of TBP will be used in the numerical experiment in Section 3.5.3:

Maximum workload TBP: the difference between the number of customers who need service from station j and the number of customers who need service from the busiest station still required by customer i, i.e., $\delta(u_1, ..., u_m, i, j) = \max_{l \in S_i^U} u_l - u_j$.

Of course, there are many other possible specifications of TBP. For example:

Maximum workload Kanban: the number of customers who need service from the busiest station still required by customer i, i.e., $\delta(u_1, ..., u_m, i, j) = \max_{l \in S_i^U} u_l$.

For stations with more than one server, we can also consider the number of servers in the specification:

Normalized Maximum workload TBP: the difference between the number of customers who need service from station j and the number of customers who need service from the busiest station still required by customer i normalized by the number of servers at the respective stations, i.e., $\delta(u_1, ..., u_m, i, j) = \max_{l \in S_i^U} \frac{u_l}{n_l} - \frac{u_j}{n_j}$.

It is also possible to normalize either of the above rules by the expected service time required at each station. Moreover, we may also take precedence constraints into account by considering the location of each customer in the process and how likely they are to affect the workload of station j during the relevant period. For example, u_j can be calculated as the number of customers who still need service from station j and have finished all station j's precedent stations. However, exact rules that quantify how likely these customers to affect the workload at station j and what is the relevant period are not simple. Thus, we do not consider such rules when inserting SI.

3.4 Case Study - Data Collection and Analysis

In the previous section, we defined a number of different dynamic scheduling policies (DSPs) for a stochastic open shop and showed how they can be modified to incorporate the threshold-based strategic idleness delays. We next apply these ideas to the case of the medical clinic operated by XYZ Inc. The background of this case study and the data used in our analysis is described in the current section.

3.4.1 Company Background and Description of Data

The clinic operated by XYZ consists of up to 21 different medical tests. These include a series of routine diagnostic tests, including blood and urine lab tests, chest X-ray, Abdominal Ultrasound, Fitness Test, Treadmill Test, Physician Exam, Audio Visual Test, Nutrition and Review with doctors. While the stations above are required by almost every customer, there are add-on assessments that can be requested, such as Optometry, Echocardiogram, Genetic Risk Assessment, etc. Each test is conducted at a specific station with possibly multiple (up to 8) servers.

On average, each customer visits 10 stations, including all nine routine diagnostic tests and one add-on. The incoming customers are directed through the process by specially trained schedulers. Every time a customer finishes a test she is led to the waiting room from which she will be picked up for the next test.

As discussed above, XYZ considers three SLMs as their objectives: the expected average total system time, the expected number of tardy customers, and the number of red faces, given by (3.1-3.3), respectively. The company's goal is to complete the service in four hours (i.e., in (3.2) $T^o = 4$ hours) and a "red face" is defined as a wait exceeding 20 minutes at any given station (i.e., in (3.3) $T^s = 20$ minutes).

We first focus on obtaining a detailed picture of XYZ's actual service and waiting time performance. We obtained two months of data from XYZ. For each customer visit, the data contains the basic information of the appointment, such as the customer's appointment time, arrival time, and departure time. For each specific test, the data also contains the customer number, station number, starting time, ending time, and service time. During these two months,

there were 41 business days, in which just over 2000 customers visited the clinic, and about 24,000 tests were performed. The number of customers who visited the clinic each business day ranged from 25 to 61 (with a mean of 49 and a standard deviation of 7.2).

3.4.2 The IT System at XYZ

XYZ's main selling point is convenience: instead of having to wait for weeks or months for all tests to be scheduled and performed, the customer can have the full assessment completed in a matter of a few hours, and have the results reviewed by the doctor who has a comprehensive view of the customer's current state of health. XYZ's clients are mostly busy executives and professionals who are willing to pay several thousand dollars for this convenience.

Not surprisingly, XYZ is extremely customer-focused, promising to complete the assessment in four hours. "Red face" incidents are regarded as significant service failures and are carefully tracked by the IT system. When a customer's wait reaches 15 minutes at any station, an anxious "yellow face" appears on the schedulers' screen, alerting them to bring this customer into service. When a customer's wait reaches 20 minutes, an angry "red face" flashes on the scheduler's screen, which typically triggers an immediate response - the customer is offered an apology, and the customer's service is expedited as much as possible. The number of red faces that occur during each day is tracked and used as part of performance reviews for process schedulers and their supervisors.

3.4.3 Operational Procedures at XYZ

XYZ schedules the arrivals of its customers at different times throughout the morning of each day. The clinic opens at 7:00am, and closes when all customers leave, typically around 16:00pm. Figure 3.1 illustrates the histograms of average daily scheduled arrivals alongside the histogram of average daily realized arrivals. Note that, the resulting arrival pattern is not stationary (contrary to the assumptions of traditional queueing models). On average, XYZ schedules 5 customers every half hour starting from 7:00am and typically until 10:30am. Then, from 10:30am to 12:00pm, the number of scheduled arrivals is gradually reduced.

We next investigate the order of service. While the order in which services are performed differs by customer, some dominant flows can be ascertained. To this end, we define $p_{i,j}$ as the

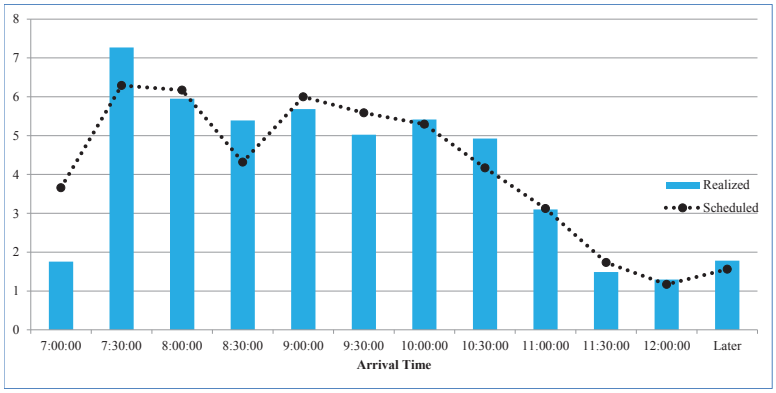

Figure 3.1: The Histogram of Average Daily Arrivals.

probability that a customer visits station i before station j, given that station i and j are both visited by this customer:

$$p_{i,j} = \frac{\text{Number of times station } i \text{ is visited before station } j}{\text{Total number of visits containing both station } i \text{ and } j}.$$

For example, suppose that three customers, A, B and C, visit the clinic. Customers A and B visit station i before station j, while customer C visits station j before station i. In this example, we have $p_{i,j} = 67\%$, $p_{j,i} = 33\%$.

Table 3.1 presents p_{ij} for the nine routine stations. These values allow us to identify some dominant work flows For example, the 98% in the first row (Lab Work) and second column (Abdominal Ultrasound) means that 98% of the time Lab Work is completed before the Abdominal Ultrasound.

Note that $p_{i,j} + p_{j,i} = 100\%$ may not always hold, because of the occasional need to re-do a test (this affects less than 1% of all customers). The same reason causes the non-zero items on the diagonal. In addition, some customers may choose not to perform all tests, resulting in $p_{i,j} + p_{j,i} < 100\%$ for some stations i and j.

There are three main observations from Table 3.1:

1. The two tests Lab Work and Abdominal Ultrasound are visited before any other stations in almost all cases. Since these two tests need to be performed on an empty stomach, the schedulers attempt to put them at the beginning of each customer's visit, so that

	Labwork	Ab Ultra	PhyExam	Treadmill	Nutrition	FitnessTest	Xray	AuViTest	ReviewDr.
Lab Work	1%	98%	99%	99%	100%	100%	95%	100%	100%
Ab Ultra	3%	1%	73%	84%	100%	98%	82%	96%	100%
PhyExam	2%	27%	1%	69%	64%	68%	59%	69%	100%
Treadmill	1%	16%	29%	1%	59%	67%	55%	73%	99%
Nutrition	1%	0%	34%	37%	0%	56%	49%	58%	82%
FitnessTest	1%	3%	31%	32%	40%	1%	44%	52%	85%
Xray	5%	14%	31%	35%	38%	44%	9%	46%	63%
AuViTest	1%	5%	30%	26%	37%	46%	41%	1%	93%
ReviewDr.	0%	0%	0%	0%	13%	14%	19%	7%	1%

Table 3.1: The Service Order Matrix

customers can have a snack as soon as possible. Note that Lab Work, which is faster, is done before Abdominal Ultrasound 98% of the times.

2. The procedure "Review with a Doctor" is typically done after the customer finishes most tests. In this station, the doctor receives reports from all other stations and thus have a comprehensive view of customer's health situation. Although customers are given the option to skip this step and receive the test results via email, most of XYZ's customer choose to attend this station.

3. The order of customers' visits to all other stations is quite random - substantiating the view of this system as a open shop with only a few precedence constraints.

In view of these three observations, the network operated by XYZ can be loosely separated into three parts: starting with Lab Work, Abdominal Ultrasound and breakfast; continuing with the other required or optional tests in some random order; and concluding with reviewing results with the doctor. The network is illustrated on Figure 3.2; all three triangles marked with "W" represent the same centralized waiting room.

3.4.4 Waiting Time Distribution

Figure 3.3 illustrates the histogram of waiting times with bin interval of a minute. The label above each column represents the relative frequency of waiting times in the bin $[t-1, t)$, for $t = 1, 2, ..., 30$. The mean and standard deviation of the waiting times are 5 and 7 minutes respectively.

Observe that the frequency of W decreases with t smoothly, except at two points: $t = 16$ and 21. At $t = 16$, the histogram of W breaks the decreasing pattern and showing an unusual

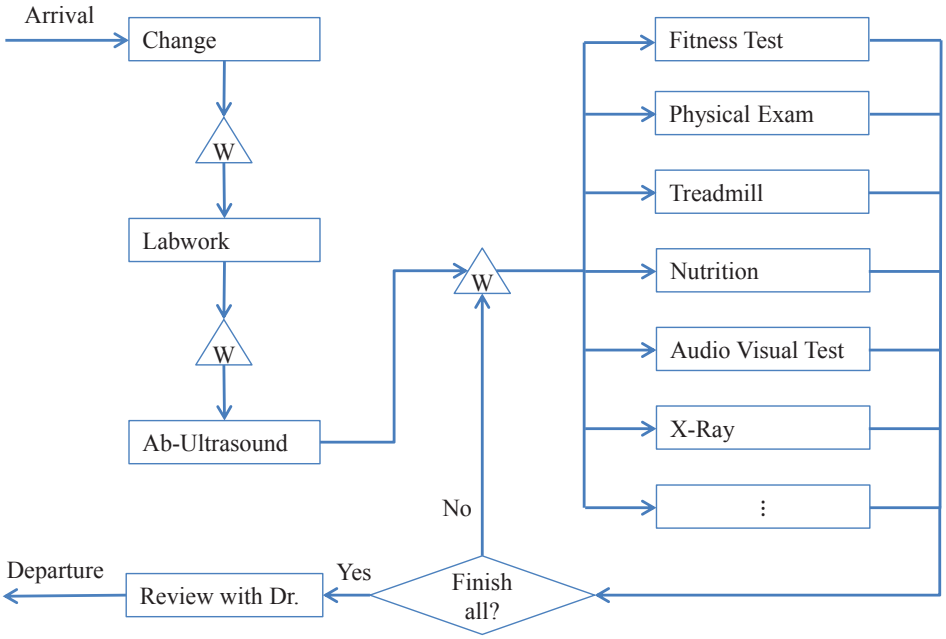

Figure 3.2: Typical Service Order at XYZ.

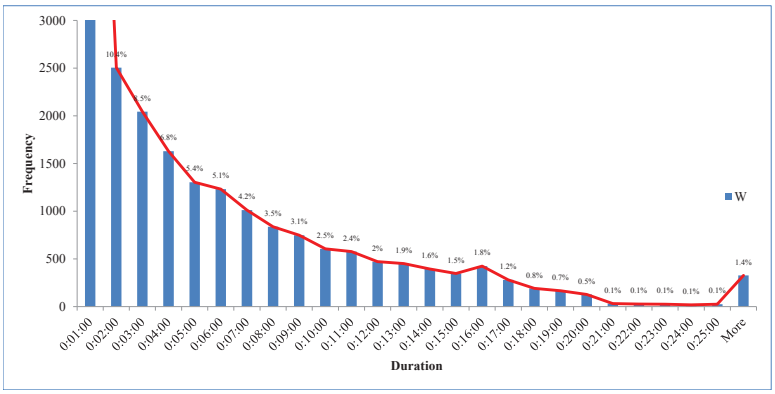

Figure 3.3: The Histogram of Waiting Times.

peak. At $t = 21$, the histogram decreases steeply to 0.1% (from 0.5% at $t = 20$), and it stays at a fixed level 0.1%. These two sudden changes in Figure 3.3 appear to be related to the appearance of yellow faces (at 15 minutes) and red faces (at 20 minutes) on the schedulers' screen.

The schedulers try their best to keep the number of red faces low. Our hypothesis is that these sudden changes in the histogram of W are reflections of these two alarm signals. No special action is taken before the first alarm. Once a yellow face appears, schedulers attempt to expedite, causing density concentration at 15 minutes. From this point, the customer is watched very carefully, and, for the most part, not allowed to get beyond 20 minutes. This causes the dip at 20 minutes. Discussions with the personal at XYZ further supports this hypothesis.

These observations demonstrate that the schedulers manage the waiting time to control not only the macro-level SLMs, like the total system time, but also the micro-level SLM, i.e., the number of red faces.

3.4.5 Performance Analysis

In this section, we analyze the performance of XYZ's open-shop network focusing primarily on the nine routine stations. To calculate the utilization of each station, we first focus on each of its servers, and calculate the average starting time (when the first customer arrives) and the average closing time (when the last customer leaves). Then, for each station, based on the statistics of its servers, we derive the average time span (the time between starting and closing) and the average busy time in it. At last, each station's utilization is obtained as the ratio of its average busy time and its average time span. We attribute the waiting time to the station that immediately followed the wait.

Table 3.2, sorted by utilization, summarizes the performance of these nine routine stations over the two months. The table also provides the three main SLMs and average total waiting time.

Based on the empirical data, the service levels were already quite good. The current average total system time is just over four hours. The average total waiting time is about an hour, i.e., less than a quarter of the total time a customer spends in the system. The incidence of red faces (waits more than 20 minutes) is 456: with 10 stations per visit on average (9 routine

Stations\Ave.	# Servers/day	Waiting Time	Service Time	Utilization	# Red Faces
FitnessTest	7.4	5:46	20:28	78%	45
PhyExam	7.7	4:50	32:45	73%	59
Doc Review	7.7	7:06	14:36	73%	173
Treadmill	4	4:48	21:50	72%	20
Ab Ultra	4	5:23	16:16	71%	10
Nutrition	3.9	4:42	19:49	69%	18
AuViTest	3.9	6:57	16:51	63%	55
Xray	1	5:23	6:14	50%	6
Labwork	1	4:19	3:23	48%	2

Ave.SystemTime	4:04:26
SystemTime≥4hrs	52.5%
Ave.TotalWaitTime	1:01:08
# Red Faces	456

Table 3.2: Summary of CHA Stations in the Empirical Data.

stations and 1 "add-on" station), this corresponds to less than 2.5% of all tests and about 25% of customers experiencing a "red face".

From Table 3.2, we see that the overall utilization, between 48% and 78%, and the waiting time, averaging between 4-7 minutes per station, are not high. In a service network with moderate variability (observed coefficient of variations were between 0.24 and 0.6 and close to 0.5 on average), we expect relatively low utilizations to be required to maintain short waiting times. This confirms the customer-centric emphasis of XYZ.

We also observe that Review with Doctor, Fitness Test, and Audio Visual Test account for most of red faces incidents; these stations also have the longest average waiting times. The Review with Doctor is the one contributing the largest number of red faces (38% of the red faces). Although it seems like a good idea for XYZ to hire more doctors to improve the SLMs, the cost of this action may be prohibitive.

The Fitness Test and Audio Visual Test together generate 100 red faces (21%) in total. The main problem at these two stations appear to be late starting times. On average, the Fitness Test and the Audio Visual Test start 2.5 hours and 1.5 hours, respectively, after the clinic opens. Moreover, they are often not at full capacity as some operators arrive even later.

We note that while the Fitness Test and Review with Doctor may be considered bottleneck stations of the network, since they both have high capacity, utilization and large associated waiting times. The Audio Visual Test, with a relatively low utilization, does not meet the standard definition of a bottleneck station. Therefore, for lack of a better name, we call Fitness

Test, Review with Doctor, and Audio Visual Test *problematic* stations, and other stations "non-problematic".

3.4.6 Evidence of Strategic Idleness in Practice

As discussed earlier, Baron, et al. (2013) demonstrated analytically, in a two-station tandem queue network, that a scheduling policy incorporating Strategic Idleness (SI) might be helpful in reducing the number of red faces. The basic idea is that when a long queue accumulates at the downstream station, it may be better to idle the upstream stations to allow the downstream queues to dissipate. However, in service operations, we are not aware of any empirical evidence of the use of SI in practice. As discussed below, the data obtained from XYZ strongly suggests that this technique is, in fact, practiced by XYZ's process schedulers (largely without the knowledge of management).

From the empirical data provided by XYZ, we discovered a number of instances where a customer was waiting for a station that was free and waiting for this customer. In other words, certain part of customers' waiting time is spent waiting for an idle station that appears to be waiting for this customer; we call such period an Overlapped Waiting (OW) time.

Initially, we thought that the existence of OW is a data error, but OWs are quite abundant and accompany 78.7% of services. Figure 3.4 depicts the histogram of the OW with bin intervals of one minute. The average OW is 2.5 minutes with a Standard Deviation of 4.5 minutes. About 50% of the OWs are less than one minute. However, 16.3% of OWs are more than five minutes, i.e., more than the average waiting per station.

One explanation is that OW might be a result of routine procedures, like room cleaning, writing reports, etc. However, as verified by our partner at XYZ, room cleaning or report writing typically do not take that long, so while these may explain the shorter OWs observed, they do not explain OWs of over 1-2 minutes.

As discussed earlier, SI can be an effective strategy for reducing the occurrence of long waits within the process. Thus given the importance placed by XYZ on minimizing the number of "red faces", an alternative explanation is that longer OWs provide the evidence of the usage of SI by the schedulers at XYZ. Of course, the mere presence of OW does not necessarily indicate that SI is being used. However, the prevalence of longer OWs in situations where SI

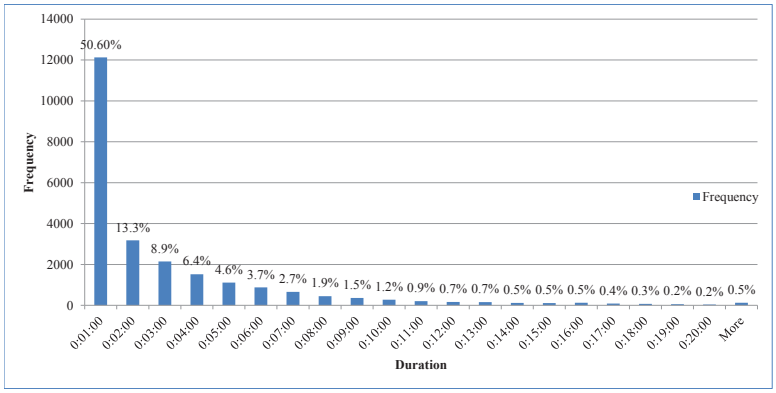

Figure 3.4: The Histogram of Overlapped Waiting.

policy would be most useful does provide support for this hypothesis. Below, we use statistical hypothesis testings to demonstrate that the timing of longer OWs provides evidence for the usage of SI at XYZ.

Statistical Evidence of the Use of Strategic Idleness

Based on our intuition and the earlier discussion, if SI is indeed used, we would expect this to be reflected in OWs as follows. When station A is ready to serve customer i, and customer i's next station is station B, which is congested, station A would use longer OWs to balance customer i's waiting time at both stations. The congestion at station B can be indicated by the fact that station B is a problematic station (Fitness Test, Review with Doctor, and Audio Visual Test), or that station B's previous customer (served just before customer i) experienced a long waiting time (e.g., ≥ 15 minutes) just before entering station B. In the later case, we say that station A precedes a "potential long wait". Similar logic suggests that to avoid wasting capacity in problematic stations, these stations should use shorter OWs than other stations. Specifically, we anticipate:

1. OWs at problematic stations are shorter than OWs at non-problematic stations;

2. OWs at stations preceding problematic stations are longer than OWs at stations preceding non-problematic stations;

3. OWs at stations preceding "potential long waits" are longer than OWs at other stations.

The statements above can be investigated using the standard t-tests. First, we test if problematic stations have different mean OW than non-problematic stations:

Null Hypothesis (H_0): The difference between the mean OW at problematic stations and the mean OW at non-problematic stations is zero;

Alternative Hypothesis (H_A): The difference between these two mean OWs is not zero.

The two-tailed t-test (with p-value 2.9×10^{-15}) indicates that H_0 is rejected, and the non-problematic stations have significantly longer mean OW (with mean 2:42) than the problematic stations do (with mean 2:13).

Second, we test if the last station preceding problematic stations have different mean OW than stations preceding non-problematic stations:

H_0**:** The difference between the mean OW at stations preceding problematic stations and the mean OW at stations preceding non-problematic stations is zero;

H_A**:** The difference between these two mean OWs is not zero.

The two-tailed t-test (with p-value 1.3×10^{-18}) indicates that H_0 is rejected, and the stations preceding problematic stations have significantly longer OWs (with mean 2:55) than the stations preceding non-problematic stations (with mean 2:23).

Third, we test if stations preceding "potential long waits" have different mean OW than other stations:

H_0**:** The difference between the mean OW at stations preceding "potential long waits" and the mean OW at other stations is zero;

H_A**:** The difference between these two mean OWs is not zero.

The two-tailed t-test (with p-value 3.2×10^{-19}) indicates that H_0 is rejected, and the stations preceding 'potential long waits' have significantly longer OWs (with mean 3:38) than other stations do (with mean 2:09).

Thus, the statistical results strongly suggest that the schedulers are indeed attempting to reduce the number of red faces by using SI.

3.5 Evaluation of Dynamic Scheduling Policies and Strategic Idleness

The service system operated by XYZ, just as many real-life service systems, is inherently transient (each day starts with an empty system and ends after processing 49 customers on average) and it is not clear that the steady-state regime is ever achieved. Since analytical methods run into significant difficulties when analyzing transient behavior, we developed a simulation model of XYZ to gain better understanding of the system and to test the performance of different policies described in Section 3.3.3 and 3.3.4 above. The simulation model was implemented in MATLAB and validated using the empirical data. Specifically, we used real service and arrival times in the model. With the simulation model, we are able to: 1) Analyze the transient behavior of XYZ's service network. By simulating the system operations over one day for 100 times (from arrival of the first customer and until the departure of the last customer at the end of the day), we can ensure that the distribution of system's performance is captured by the simulation model. 2) Simulate scheduling decisions that depend on the state of the system. Both the customer and station assignments rules can be replaced with one of the policies described in Section 3.3.3. 3) Measure the macro-level and micro-level SLMs: the expected average total system time, the expected probability of total system time longer than four hours, and the number of red faces. 4) Simulate the performance of scheduling policies that incorporate strategic idleness (as described in Section 3.3.4).

The simulation recorded the resulting three SLMs for each scheduling policy. For each working day, we keep the available resources (station availabilities and opening times), customers' service needs and customers' service times at different stations the same as in the real data, and only change the scheduling policy.

In Section 3.5.1, we investigate the performances of DSPs without SI modifications (work-conserving) to evaluate the benefit of automated DSPs with respect to the global performance measures. In Section 3.5.2, we compare the performance of the actual scheduling policies used by XYZ with and without OWs. Next in Section 3.5.3, we compare DSP+TBP policies with the actual ones.

3.5.1 Performance of Work-conserving Dynamic Scheduling Policies

The simulated performance of various dynamic scheduling policies described in Section 3.3.3 is presented in Table 3.3. The third row of the table contains the Empirical Data (ED), i.e., the results for the actual scheduling policy used by XYZ. We start by comparing the various Dynamic Scheduling Policies (DSPs) without the Strategic Idleness (SI) modification introduced in Section 3.3.4 - we call these the "non-idle" versions of the respective policies. The results for all policies can be found in the second block of the table.

We first observe that the Shortest Expected Remaining Processing Time first (SERP) policy dominates the LAW, LCW, and LERP policies: SERP has a lower average total system time, a lower probability of spending more than four hours in the system, and a lower number of red faces. Comparing SERP policy with the Longest System time first (LS) policy shows that LS outperforms SERP with respect to the number of red faces, while SERP policy performs better on the other two SLMs.

We also observe that all of the DSPs outperform the actual (ED) scheduling policy with respect to both macro-level SLMs used by XYZ: they reduce the average total system time by about 40 minutes (16%), and the proportion of customers experiencing total system times of over four hours from 52.5% to around 21%.

However, for all DSPs the incidence of red faces is substantially higher than for the ED policy. Even the best-performing Longest Mean Overage Processing time first (LMOP) policy experiences an increase in this measure by 34.2% to 612 vs. the ED policy - a clearly undesirable outcome. We suspect that the reason that the DSPs we proposed perform poorly with respect to the number of red faces measures is that they are work-conserving (non-idle) policies, while, as discussed in Section 3.4.6, it appears that the XYZ's schedulers are actively using Strategic Idleness to manage the number of red faces. We further investigate this issue in the following section.

3.5.2 Effect of Overlapped Waiting Times: Another Indication of Strategic Idleness

To investigate the usage of Overlapped Waiting (OW), we define the "non-idle" version of the actual scheduling policy by eliminating the OWs. For example, suppose customer A finishes service at station i. To find the next customer for station i, we examine the data and find that customer B is the next customer of station i. If customer B is in the waiting room and station i is her next station (based on data), she is immediately taken to station i; otherwise, station i stays idle until the arrival of customer B. We follow a similar rule when choosing the next station for customer A. We call this non-idle version of the actual scheduling policy "ED+Non-idle". The corresponding results are presented on row 4 of Table 3.3.

Our simulation results indicate that if the clinic was operated under ED+Non-idle policy during the two month in our data, the average total system time would drop by 18 minutes to 3:46:37, and the proportion of customers with system time of over four hours would decrease by 14.3% to 38.3%. Thus, both of the macro-level SLMs would improve substantially (though the improvements still fall far short of those observed under the DSPs described earlier). However, the total number of red faces would increase to 1094 (140% increase). Out of the 456 red faces observed in the data, 217 disappear in the simulation under the ED+Non-idle policy, while 855 new ones emerge. Shorter waiting time at previous stations, i.e., elimination of SI in the form of OW, causes 705 of these new red faces.

The mean and standard deviation of the waiting time from the simulation result for ED+Non-idle policy are 3.5 minutes and 7.5 minutes respectively. Figure 3.5 shows the histogram of W (waiting time) for the ED+Non-idle policy alongside the histogram for the ED policy. We see that the histogram of W from ED+Non-idle policy is much smoother than the histogram from the actual scheduling policy. Those odd jumps at the $t = 16$ or 21 minutes observed in the empirical data disappear, while the number of waits of more than 15 minutes is at the same level (about 2000) for both ED and ED+Non-idle policies.

Comparing the ED+Non-idle policy with the ED policy indicates that by intentionally holding back customers at non-problematic stations when a customer is likely to experience a long wait at the problematic station downstream, the ED policy effectively re-distribute

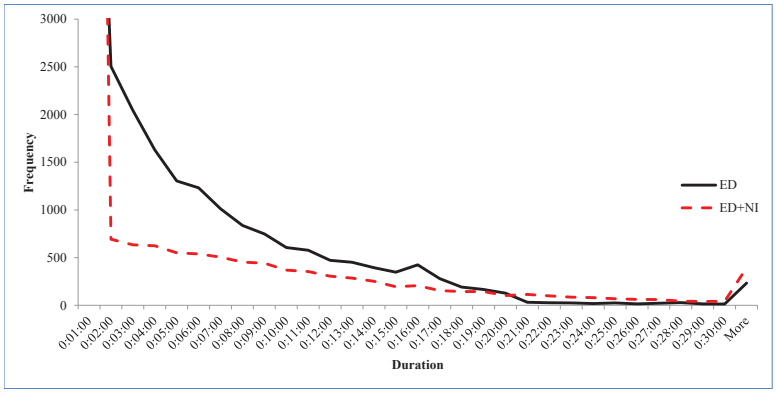

Figure 3.5: Histogram of Waiting Times under Different Scheduling Policies.

the waiting times more evenly within the network. These observations further support the conclusions from our statistical hypothesis tests that XYZ's schedulers are using SI.

To summarize, the results above indicate that the reason that ED policy outperforms those proposed policies in the number of red faces is that these policies are non-idle policies, while the XYZ's schedulers are using SI in their scheduling policy. However, the schedulers of XYZ are using their own intuitions to insert OW. There were no official policies to this effect - as discussed earlier, the upper management seemed to be unaware of this practice.

In the following section, we introduce SI into our proposed DSPs by using the Maximum Workload Threshold Based Policy (TBP) as discussed in Section 3.3.4 earlier.

3.5.3 Performance of DSPs with Strategic Idleness Modification

In this section, we add the SI to the different DSPs. Specifically, we use the generalized TBP modification to the LAW, LS, LMOP, LCW, SERP and LERP policies. We demonstrate that this modification can reduce the number of red faces substantially without a significant deterioration in the values of the macro-level SLMs. We employed the "Maximum Workload TBP", described in Section 3.3.4, to the six proposed DSPs, and implemented the Overtake-free SI. This ensures that, after the SI period, customers would not be delayed once station resumes working. The results can be found in the third block of Table 3.3.

Comparing the second block to the third block reveals that in all six DSPs, the SI policy results in a small 5-7% increase in total system times (by about 11 minutes in the LAW, LS,

	System Time			Red Faces				
Policies \ Measures	Mean	Stdev	≥4hrs	#(≥ 15mins)	#(≥ 18mins)	#(≥ 20mins)	#(≥ 22mins)	#(≥ 25mins
Empirical Data	4:04:26	1:00:51	52.5%	1645	749	456	397	327
ED+Non-idle	3:46:37	56:53	38.3%	1846	1340	1094	880	645
LAW+Non-idle	3:23:53	49:55	21.3%	958	822	730	661	541
LS+Non-idle	3:24:21	47:44	21.3%	873	743	665	607	520
LMOP+Non-idle	3:24:28	52:01	20.4%	807	683	612	543	457
LCW+Non-idle	3:24:19	50:07	21.2%	972	816	729	653	554
SERP+Non-idle	3:23:17	50:57	20.6%	929	784	687	606	507
LERP+Non-idle	3:25:22	50:58	20.9%	926	791	705	633	528
LAW+MaxWrkldTBP	3:34:32,	51:46	28.0%	828	647	551	477	393
LS+MaxWrkldTBP	3:35:03	51:29	28.2%	841	622	518	444	360
LMOP+MaxWrkldTBP	3:35:55	53:40	28.9%	707	539	476	427	355
LCW+MaxWrkldTBP	3:37:17	52:54	30.7%	845	640	551	478	380
SERP+MaxWrkldTBP	3:34:25	52:35	28.2%	792	618	525	464	380
LERP+MaxWrkldTBP	3:38:58	53:16	31.7%	878	692	609	541	445
LAW+MaxWrkldKB	3:43:19	52:26	35.6%	875	683	567	481	385
LS+MaxWrkldKB	3:43:14	50:43	34.5%	829	622	538	459	370
LMOP+MaxWrkldKB	3:37:37	52:02	29.2%	745	593	520	465	377
LCW+MaxWrkldKB	3:43:14	50:43	34.5%	885	687	588	506	428
SERP+MaxWrkldKB	3:43:15	53:00	34.5%	904	671	557	461	358
LERP+MaxWrkldKB	3:40:09	53:07	30.8%	887	747	648	574	476

Table 3.3: Performance of Different Scheduling Policies with and without SI.

LMOP and SERP policies, and about 13 minutes in LCW and LERP policies) and 8-10% increases in the probability of experiencing total system times of over four hours versus the non-idle version of these policies. However, the red face measure (i.e., the number of waits over 20 minutes) is reduced by around 24%. This is in line with the theoretical analysis presented in Baron et. al. (2013): the SI policy is able to significantly reduce the probability of long waits (an equivalent measure to red faces), while only slightly increasing the total system times. Note that the SERP+TBP policy again dominates LAW+TBP, LCW+TBP and LERP+TBP policies.

While the numbers of red faces under our DSPs with TBP modification are greater than under the ED policy (476 vs. 456), a significant reduction in total system times, as well as an automated DSP without any supervision presents an attractive trade-off to the decision-maker. Incidentally, the difference in the number of red faces versus the ED policy may be due to the fact that in real system expediting action appears to be taken by process managers somewhat before the waiting time reaches 20 minutes (which is natural, given that red faces incidents are used in performance reviews). If we redefine "excessive waits" to be 15 or 18 minutes, rather than the current 20 minutes, the incidence of such waits under all six DSP+SI policies falls below that in the empirical data.

Our main finding from this comparison is that the use of strategic idleness in conjunction with the dynamic scheduling policies, such as our LS, LAW and SERP policies, can be used to improve the SLM in practice. Moreover, due to the simple and transparent structure, the cost of implementing such policies should be low. In addition to supporting the main theme of the paper on the joint usage of DSPs and SI, this finding is also encouraging for the usage of decision support systems for managing service network in practice.

3.5.4 Comparison of Generalized Threshold Based Policy to Kanban Policy

In this section, we compare the TBP with the Kanban policy - a widely applied policy in manufacturing setting. Specifically, we use the Maximum Workload Kanban policy, described in Section 3.3.4, to add SI to different DSPs and compare the result (the fourth block in Table 3.3) with the performance of these DSPs with TBP modification (the third block in Table 3.3).

The result suggests that, when combined with any DSP, the TBP clearly dominates the Kanban policy. This is in line with Baron et al. 2014 that shows that the Kanban policy is less effective when different stations have similar service rates and there is no clear bottleneck. The Kanban policy focuses only on the waiting at the bottleneck but ignores waits at other stations. Thus, it performs well in systems with a simple structure when the bottleneck is clear. In the case of XYZ, the TBP for introducing SI has more flexibility and therefore dominates the Kanban policy in all of these examples on all the measures depicted in Table 3.3.

3.6 Summary and Open Questions

In this paper we investigated the performance of Dynamic Scheduling Policies in a stochastic open-shop service network. The unique feature of the system we examined is the need to balance between the more traditional "macro-level" service level measures, such as the total system time, and the customer-focused "micro-level" measure related to excessive waits within the system. The incidence of excessive waits can be managed by introducing "strategic idleness" where intentional (small) waits are introduced in upstream stations to prevent (longer) waits at busy downstream stations. Our work was motivated by the data from a real-life medical clinic. Through process analysis and statistical hypothesis tests we demonstrate that (largely

unbeknownst to management), system schedulers appear to use strategic idleness to minimize instances of excessive waits.

We developed a flexible framework allowing us to represent "completely reactive" dynamic scheduling policies by defining simple scoring rules. We also showed how a given policy can be modified with a "strategic idleness" component allowing it to account for both micro- and macro-level measures. By developing a simulation model based on the real data for the XYZ system we showed that these automated scheduling policies appear to be quite promising: achieving substantial improvements on the macro-level measures while essentially matching the performance of actual policies on the micro-level measure.

Due to the complexity of the underlying system, our results are mostly computational. Analytical substantiation of some of our conclusions would be quite interesting. A step in this direction was taken by Baron et al. (2013) who investigated policies with strategic idleness analytically for a tandem queue. An extension of their results to more complex stochastic networks remains open.

References

Afèche, P. (2013) Incentive-compatible revenue management in queueing systems: optimal strategic delay. Manufacturing Service Oper. Management Forthcoming.

Alcaide, D., J. Sicilia, D. Vigo. (1997) A tabu search algorithm for the open shop problem. TOP (Trabajos de Investigación Operativa), 5 (2) (1997), pp. 283–29.

Alcaide, D., A. Rodriguez-Gonzalez, J. Sicilia. (2006) A heuristic approach to minimize expected makespan in open shops subject to stochastic processing times and failures. Int J Flex Manuf Syst, 17 (3) pp. 201–226.

Baron, O., O. Berman, D. Krass. (2008) Facility Location with Stochastic Demand and Constraints on Waiting Time. *Manufacturing & Service Operations Management* Vol. 10, #3, pp. 484-505.

Baron, O., O. Berman, D. Krass., J. Wang (2013) Using Strategic Idleness to Improve Customer Service Experience in Service Networks. *Operations Research Forthcoming.*

Baron, O., J. Milner. (2009) Staffing to Maximize Profit for Call Centers with Alternate Service

Level Agreements. *Operations Research*, 57, pp. 685-700.

Bertsimas, D., D. Gamarnik, J. Sethuraman (2003) From Fluid Relaxations to Practical Algorithms for High-multiplicity Job-shop Scheduling: The Holding Cost Objective, *Operations Research*, 51(5), 798-813.

Bouch, A., A. Kuchinsky, N. Bhatti. (2000) Quality is in the eye of the beholder: Meeting users' requirements for Internet quality of service. In \emph{Proc. of CHI2000 Conference on Human Factors in Computing Systems}, ACM Press, pp. 297-394.

Dai, D., W. Lin (2005) Maximum Pressure Policies in Stochastic Processing Networks, *Operations Research*, 53(2), 197-218.

de-Véricourt F., Y.-P. Zhou (2005) A routing problem for call centers with customer callbacks after service failure. *Operations Research* 53(6) 968-981.

de-Véricourt F., O. Jennings. (2011) Review on Nurse Staffing in Medical Units: A Queueing Perspective. *Operations Research* Vol. 59. No. 6, pp 1320-1331, 1547-1548.

Friedman, H.H., Friedman, L.W. (1997) Reducing the "wait" in waiting-line systems: waiting line segmentation. *Business Horizons*, 40 (4), 54-58.

Gans, N., G. Koole, A. Mandelbaum. (2003) Telephone Call Centers: Tutorial, Review, and Research Prospects. *Manufacturing & service Operations Management*, Vol. 5, No. 2, Spring, pp. 79-141

Harrison, J. M. (1996) The BIGSTEP approach to flow management in stochastic processing networks. F. P. Kelly, S. Zachary, I. Ziedins, eds. Stochastic Networks: Theory and Applications. Clarendon Press, Oxford, U.K., 57-90.

Larson R. C. (1987) Perspectives on Queues: Social Justice and the Psychology of Queueing. *Operations Research* Vol. 35, No. 6 (Nov-Dec), pp. 895-905.

Maglaras, C. (2000) Discrete-review policies for scheduling stochastic networks: Trajectory tracking and fluid-scale asymptotic optimality. *Ann. Appl. Probab.* 10(3) 897-929.

Masin, M., Y. Herer, E. M. Dar-el. (2010) SWIP: A Unified Model of Self-regulating Production Control Systems. *Working paper*, Technion.

Mehrotra, V., K. Ross, G. Ryder, Y.P. Zhou. (2012) Routing to Manage Resolution and Waiting Time in Call Centers with Heterogeneous Servers. *Manufacturing & Service Operations Management* Vol. 14, No. 1, Winter 2012, pp. 66-81.

Ouelhadj, D., S. Petrovic. (2009) A survey of dynamic scheduling in manufacturing systems. J. Scheduling, vol. 12, no. 4, pp.417-431.

Pinedo, M. (1984) A Note on the Flow Time and the Number of Tardy Jobs in Stochastic Open Shops. European Journal of Operational Research, Vol. 18, pp. 81–85.

Pinedo, M., S.M., Ross (1982) Minimizing Expected Makespan in Stochastic Open Shops. Advances in Applied Probability, Vol. 14, pp. 898–911.

Pinedo, M. (2012) Scheduling: Theory, Algorithms, and Systems. Springer, New-York.

Roemer, T.A. (2006) A note on the complexity of the concurrent open shop problem, Journal of Scheduling Vol. 9, pp. 389-396.

Soman, D., M. Shi (2003) Virtual Progress: The effect of path characteristics on perceptions of progress and choice. *Management Science* Vol. 49. No. 9, pp. 1229-1250.

Saghafian S., W.J. Hopp, M.P. Van Oyen, J.S. Desmond, S.L. Kronick (2012) Patient Streaming as a Mechanism for Improving Responsiveness in Emergency Departments. *Operations Research* Forthcoming.

Smith W.E. (1956) Various optimizers for single-stage production. *Naval Res. Logist. Quart.* 3 59-66.

Stidham S. (2002) Analysis, Design, and Control of Queueing Systems. *Operations Research* **50**(1) 197-216.

Taylor, S. (1994) Waiting for service: the relationship between delays and evaluation of service. *Journal of Marketing*, 58 (April), 56-69.

Van Mieghem, J.A. (1995) Dynamic Scheduling with Convex Delay Costs: the Generalized $c\mu$ Rule. *Annals of Applied Prob.* 5(3) 809-833.

Chapter 4

$M/M/c$ Queue with Two Priority Classes

4.1 Introduction

In many industries there is a growing usage of prioritization: companies often prioritize groups of customers in order to improve market segmentation, service, and profitability. For example, emergency departments prioritize more seriously injured patients; websites prioritize paid users over free ones; and car rental companies prioritize customers with reservations over "walk-in"" customers. A key consequence of this prioritization is that different customer classes experience different response (sojourn) times, the time from the arrival of a customer until her departure. (A related measure is the waiting time, the time from a customer's arrival until her service begins, whose distribution can be derived from the distribution of response and service times.) Not surprisingly there is a vast literature that investigates prioritization in queueing systems. Much of this literature uses queueing theory to derive and analyze different prioritization policies in services (i.e., Maglaras and Zeevi 2005 and references therein), inventory settings (i.e., Abouee-Mehrizi et al. 2012 and references therein), and dynamic scheduling (i.e., Van Mieghem 1995 and references therein). This literature typically focuses on characterizing the distribution of the response time of different priority classes.

While this literature has been able to establish the distribution of response times for single-

server priority queues such as the $M/G/1$ (see i.e., Takagi 1991), finding this distribution is much more difficult in the multi-server setting. Much of the multi-server literature has focused on the $M/M/c$ queue. The $M/M/c$ queue with multiple priority classes was first investigated by Davis (1966), who considered a non-preemptive system with the same service rate for all priority classes, finding a closed-form expression for the Laplace Transform (LT) of any priority class's waiting time. For the same setting, Kella and Yechiali (1985) elegantly derived the LT. Buzen and Bondi (1983) gave a simple approximation for each priority class's mean response time in a preemptive system with *different* service rates for each priority class. Maglaras and Zeevi (2004) used a diffusion approximation to solve a similar problem with impatient high priority customers in a heavy-traffic regime. Finally, Harchol-Balter et al. (2005) used PH distributions to approximate the response time of a preemptive $M/PH/c$ queue with different service rates. They also provide a taxonomy of relevant literature; we refer the reader there for a more detailed literature review.

To the best of our knowledge, no exact solution for the response time distribution in a multi-server queueing system serving multiple priority classes with *different* service rates has appeared in the literature. In this paper, we consider the $M/M/c$ queue with two preemptive priority classes. Class-i jobs arrive according to a Poisson process with rate λ_i, $i = 1, 2$. Service times for Class-i jobs are exponentially distributed with parameter μ_i, $i = 1, 2$. For stability, we require $\sum_{i=1}^{2} \frac{\lambda_i}{\mu_i} < c$. (We assume preemptive resume for Class-2 jobs.)

This paper's main contribution is to characterize the Generating Function (GF) of the distribution of the number of Class-2 jobs in steady state: we derive a closed-form expression for this GF for $c = 2$, and provide an exact numerical method to calculate this GF for $c > 2$. Since that Class-1 jobs have preemptive priority, their analysis is straightforward.

We derive these GFs using a new approach for the analysis of continuous-time Markov Chains (MCs): the main difficulty in analyzing the $M/M/c$ queue with two priority classes is the need to track the number of jobs from each class. Thus, the state of the system is expressed as a 2D-infinite continuous-time MC. We apply a new method to simplify this MC to a 1D-infinite discrete-time MC that is more tractable. We then analyze this discrete-time MC by observing the system state embedded at Class-2 departures, expressing these using a similar process to the one used in analyzing the standard $M/G/1$ queue. Since the complexity of

deriving the GF increases with the number of servers, c, we also provide a numerical algorithm to derive this GF for $c > 2$. Using this algorithm, we derive insights on how system performance changes as the characteristics of the jobs or servers change. Our methodology can also be applied to similar problems: in Section 4.7, we explain how to apply it to an $M/M/c$ queue with two priority classes, where the first class is completely impatient, as considered by Maglaras and Zeevi (2004). We believe that our methodology can be used in additional applications as well.

The paper proceeds as follows: After introducing the model and background results in Section 4.2, we present the key ideas of our methodology in Section 4.3. We demonstrate the methodology using the single-server case in Section 4.4. We provide exact expressions for the two-server case, and discuss generalization for $c > 2$ in Section 4.5. We provide an exact numerical method with good computational efficiency to solve systems with $c \geq 2$ in Section 4.6. Numerical results, insights, and extensions are given in Section 4.7. We summarize the paper in Section 8. All proofs are in the Appendix.

4.2 Model and Preliminary Results

We consider an $M/M/c$ queue with two priority classes. Let q_i, $i = 1, 2$ be the number of Class-i jobs in the system, and R_i and W_i, $i = 1, 2$ be the Random Variables (R.V.s) representing the steady state response and waiting time of Class-i jobs in the system respectively.

For the $\mu_1 = \mu_2$ case, the response time distribution of each priority class is given in e.g., Buzen and Bondi (1983). We, however, consider this problem when Class-1 and Class-2 have different service time requirements (i.e., $\mu_1 \neq \mu_2$). Figure 4.1 illustrates the MC for the number of jobs in the system and their classes. The state (q_1, q_2) represents that there are q_1 Class-1 jobs and q_2 Class-2 jobs in the system. Note that the MC is infinite in two dimensions, complicating the analysis.

From Figure 4.1, we see that Class-1 jobs' service rate is $\mu_1 \min(q_1, c)$; this service rate is independent of q_2, because Class-1 jobs have preemptive priority over Class-2 jobs. Thus, the distribution of Class-1 jobs' response time is (e.g., Section 3.4, Buzacott and Shanthikumar

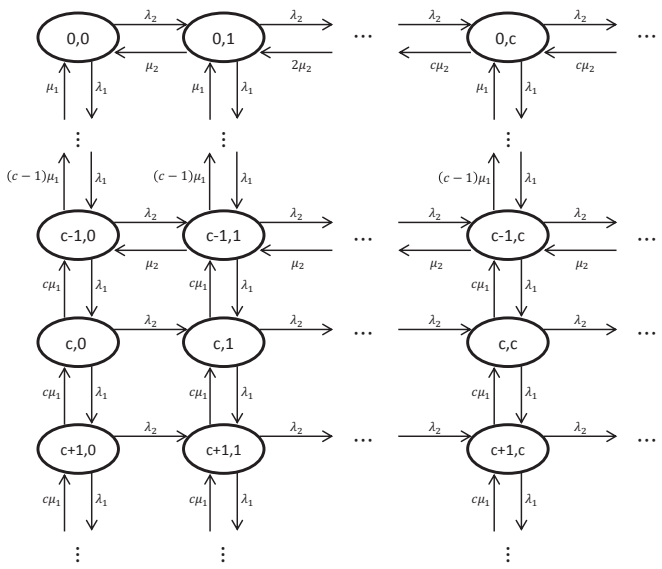

Figure 4.1: MC of the $M/M/c$ queue with two priority classes.

1993)

$$P(R_1 < t) = 1 - e^{-\mu_1 t} - \frac{(e^{-(c\mu_1-\lambda_1)t} - e^{-\mu_1 t})}{1-(c-\frac{\lambda_1}{\mu_1})} \frac{\lambda_1^c}{\mu_1^c c!}((1-\frac{\lambda_1}{c\mu_1})\sum_{i=0}^{c-1}\frac{\lambda_1^i}{\mu_1^i i!} + \frac{\lambda_1^c}{\mu_1^c c!})^{-1}.$$

Therefore, this paper focuses on deriving the response time for Class-2 jobs.

Let r_{q_1,q_2} be Class-2 jobs' service rate when the MC is at state (q_1, q_2). From Figure 4.1, we observe

$$r_{q_1,q_2} = \mu_2 \min(c - \min(q_1, c), q_2), \tag{4.1}$$

where $c - \min(q_1, c)$ is the number of servers that are free to serve Class-2 jobs when the system is in state (q_1, q_2). Let $SP(q_2) = (r_{0,q_2}, \ldots, r_{c-1,q_2})$ be the vector of Class-2 jobs' Service Pattern (SP) when there are q_2 Class-2 jobs in the system, for $q_1 = 0, \ldots, c-1$; when $q_1 \geq c$, Class-2 jobs are not served. Notice that, for any $q_2 \geq c$, we have $SP(q_2) = (c\mu_2, (c-1)\mu_2, \ldots, \mu_2)$. Before discussing Class-2 jobs, further we recall several results and define several special matrices that are used extensively in the paper.

Let t be a random time interval with LT, $LT^t(s)$; let X be the number of Poisson arrivals

with rate λ during t, and $G_X(z)$ be the GF of X. We then have:

$$P\{X = x\} = \frac{(-\lambda)^x}{x!} LT^{t(x)}(\lambda); \tag{4.2}$$

$$G_X(z) = LT^t(\lambda - \lambda z), \tag{4.3}$$

where, $LT^{t(x)}(\lambda)$ denotes the x^{th} derivative of $LT^t(s)$ evaluated at λ. Both (4.2) and (4.3) are well known results, see e.g., (3.58) and (3.67) respectively in Buzacott and Shanthikumar (1993).

Throughout this paper, we write any column vector as the transpose of a corresponding row vector to save space. Let $\mathbf{0}_{i\times j}$ and $\mathbf{1}_{i\times j}$ denote $i \times j$ matrices with all elements zero or one, respectively, and I denote the identity matrix. The following Lemma is important in Sections 4.4 and 4.5.

Lemma 2 *Assume a MC's state space is composed of two sets: a transient set, T and an absorbing set, A. Let $\Gamma_{T\to T}$ and $\Gamma_{T\to A}$ be the one step transition matrices from T to T and T to A respectively. Then, $PA_j \mid T_i$, the absorbing distribution matrix, which represents the probability that the system starts from a state $T_i \in T$ and eventually reaches a state $A_j \in A$, i.e., the probability of the system to be absorbed in state A_j once starting at state T_i, can be expressed as*

$$[P\{A_j \mid T_i\}]_{T_i\in T, A_j\in A} = (I - \Gamma_{T\to T})^{-1}\Gamma_{T\to A}. \tag{4.4}$$

4.3 Simplification - The 1D-Infinite MC

Finding the distribution of R_2 is challenging because the MC in Figure 4.1 is 2D-infinite. We apply an innovative method to simplify the 2D-infinite continuous-time MC in Figure 4.1 to a 1D-infinite discrete-time MC. This method first simplifies the system by aggregating the behavior during a *Class-1 busy period* (BP), which starts when there are c or more Class-1 jobs in the system (i.e., once q_1 increases to c) and ends when the number of Class-1 jobs drops to $c-1$ (i.e., once q_1 decreases to $c-1$).

During each BP the service rate of Class-1 jobs is $c\mu_1$ (because $q_1 \geq c$ during the entire BP) and the arrival rate of Class-1 jobs is λ_1. Thus, during this BP, the MC of Class-1 jobs

is identical to the BP of an $M/M/1$ queue with arrival rate λ_1 and service rate $c\mu_1$ (see e.g., Harchol-Balter et al. 2005). Thus, the LT of this BP is (see Takagi 1991, Chapter 1)

$$LT^{BP}(s) = \frac{1}{2\lambda_1}(\lambda_1 + c\mu_1 + s - \sqrt{(\lambda_1 + c\mu_1 + s)^2 - 4c\lambda_1\mu_1}). \tag{4.5}$$

Next, using (4.2), we express the probability of l Class-2 jobs arriving during the BP

$$\alpha_l^{BP} = \frac{(-\lambda_2)^l}{l!} LT^{BP(l)}(\lambda_2),\ l = 0, 1, 2, ... \tag{4.6}$$

Let $G_{\alpha^{BP}}(z)$ be the GF of α^{BP}; then from (4.3),

$$G_{\alpha^{BP}}(z) = LT^{BP}(\lambda_2 - \lambda_2 z). \tag{4.7}$$

During the BP, no Class-2 jobs are served; all Class-2 arrivals join the queue: When the BP is over, q_1 becomes $c-1$ and the distribution of the number of Class-2 jobs in the MC can be calculated from (4.5) and (4.6). Specifically, if the MC enters a BP from state $(c-1, q_2)$, then when the BP ends, the MC is in state $(c-1, q_2+j)$ with probability (w.p.) α_j^{BP}, for $j \geq 0$.

It is important to point out that we are not approximating Class-2 arrivals during the BP with a batch arrival at the end of the BP. Instead, we calculate the distribution of the number of Class-2 arrivals during the BP at the end of the BP. Indeed, we lose the information on when those Class-2 arrivals occurred, but we will establish next that this information is not necessary.

Using the BP, we simplify the MC: Let $v(q_1, q_2)$ denote the total rate at which the MC moves out of state (q_1, q_2). Then

$$v(q_1, q_2) = \lambda_1 + \lambda_2 + \mu_1 \min(q_1, c) + \mu_2 \min(c - \min(q_1, c), q_2). \tag{4.8}$$

Let BP_i denote a BP that started from state $(c-1, i)$, $i = 0, 1, \ldots$. After aggregating the BP's in the MC into the BP_i's, we get a 1D-infinite *discrete-time* MC with c rows: The first $(c-1)$ rows are identical to the first $(c-1)$ rows in the original MC, and the c^{th} row is composed of the BP_i's. When the MC leaves BP_i, it may enter any state $(c-1, q_2)$ with $q_2 \geq i$. Figure 4.2

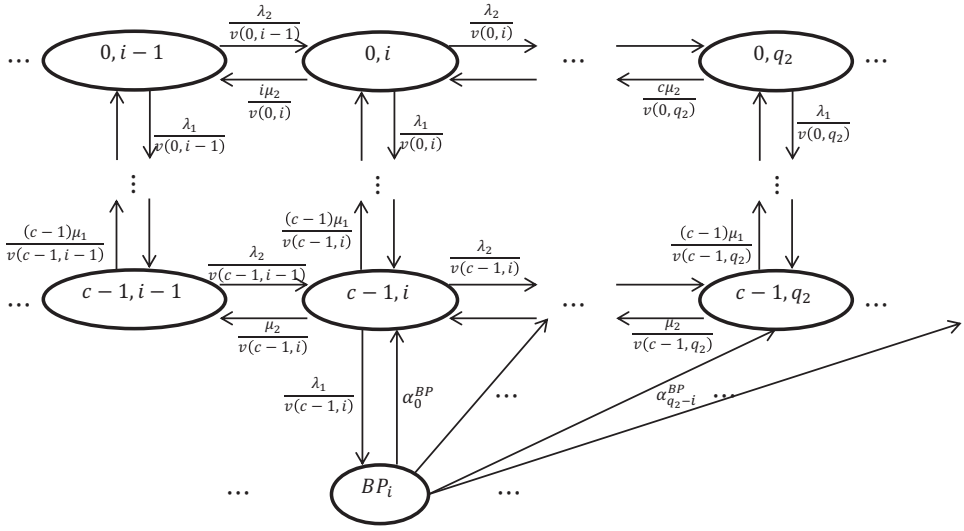

Figure 4.2: The Simplified MC.

illustrates this 1D-infinite discrete-time MC.

Still, to the best of our knowledge, there are no known closed-form solutions for this ladder-like 1D-infinite discrete-time MC. We overcome this problem by observing the system state at departure epochs of Class-2 jobs. Using the memoryless property, the distribution of the system state seen by the $(k+1)^{st}$ Class-2 departure is determined by the system state seen by the k^{th} Class-2 departure, so it is indeed a MC. This MC is called the embedded MC (EMC). To determine the steady state distribution of this EMC, we will follow the three steps used to analyze the EMC of the standard $M/G/1$ model (see e.g., Section 3.3.2, Buzacott and Shanthikumar 1993): 1) derive the one-step transition matrix of the EMC, 2) characterize the GF of the number of jobs seen by a departure in steady state, and 3) derive the unknown constant in the expression of this GF.

From the proof of poisson arrivals see time average and departures see what arrivals do in Section 5.1.3 of Gross et al. (2008), we know that the GF of the number of Class-2 jobs observed by a Class-2 departure in steady state is identical to the GF of the number of Class-2 jobs in the system in steady state, q_2. From this latter GF, the moments of q_2, such as $E[q_2]$, can be derived. Then, using Little's Law, we can calculate the expected response time of Class-2 jobs, $E[R_2]$. If we further assume that the service order of Class-2 follows the FIFO rule (e.g., when items are made to orders), we can use Distributional Little's Law from Bertsimas and Nakazato

(1995) to get the response time distribution of Class-2 jobs. This assumption is reasonable in a manufacturing setting; if c parallel machines work on orders simultaneously, when a product is finished it may be distributed to the first order in the queue.

4.4 The Single-server Case

To develop some intuition for our analytical procedure, we first demonstrate it in the single-server setting. The solution for the response time of Class-2 jobs in this case is (see e.g., Takagi (1991)):

$$LT^{\hat{R}_2}(s) = \frac{2(\lambda_1\mu_2 + \lambda_2\mu_1 - \mu_1\mu_2)}{(\mu_2 - 2\mu_1)s + \lambda_1\mu_2 + 2\lambda_2\mu_1 - \mu_1\mu_2 - \mu_2\sqrt{(s+\lambda_1+\mu_1)^2 - 4\lambda_1\mu_1}}. \tag{4.9}$$

Our methodology provides an alternative proof and, more importantly it can be used in the multi-server case. For convenience, we denote quantities related to the $c = 1$ case with a "hat" ($\hat{}$).

Let $\hat{L}_k^2$ be the state of the MC seen by the k^{th} Class-2 departure, i.e., the state of the EMC. We define the EMC's infinite dimensional transition matrix, $\hat{M}$: let the element $\hat{m}_{\hat{L}_k^2 \to \hat{L}_{k+1}^2}$ in $\hat{M}$ denote the probability that the $(k+1)^{st}$ Class-2 departure leaves $\hat{L}_{k+1}^2$ Class-2 jobs in the system, given the k^{th} Class-2 departure left $\hat{L}_k^2$ Class-2 jobs in the system, i.e., the one-step transition probability of the EMC.

We next derive an equation relating $\hat{L}_k^2$ to $\hat{L}_{k+1}^2$. Let $\hat{D}_k$ be the k^{th} inter-departure time of Class-2 jobs (i.e., the time between the k^{th} and the $(k+1)^{st}$ Class-2 departure). Let the R.V. $\alpha^{\hat{D}_k}$ be the number of Class-2 arrivals, i.e., the number of arrivals from an independent Poisson process with rate λ_2, during $\hat{D}_k$. As in the $M/G/1$ model, the number of Class-2 jobs seen by the $(k+1)^{st}$ Class-2 departure is equal to the number of Class-2 jobs seen by the k^{th} Class-2 departure minus one (the $(k+1)^{st}$ Class-2 departure) plus the number of Class-2 jobs that arrived during $\hat{D}_k$:

$$\hat{L}_{k+1}^2 = \hat{L}_k^2 - 1 + \alpha^{\hat{D}_k}. \tag{4.10}$$

From (4.10), we know that $\hat{L}_{k+1}^2 \geq \hat{L}_k^2 - 1$, so $\hat{m}_{i \to j}$ is zero, if $j < i - 1$.

Thus, the transition matrix has the form illustrated in (4.11). Each row and column is

labeled by the corresponding state $\hat{L}_k^2$. All elements of the lower triangle below the second row in $\hat{M}$ are zero.

$$\hat{M} = \begin{array}{c|ccccc|c} & 0 & 1 & 2 & 3 & 4 & \cdots \\ \hline 0 & \hat{m}_{0\to0} & \hat{m}_{0\to1} & \hat{m}_{0\to2} & \hat{m}_{0\to3} & \hat{m}_{0\to4} & \cdots \\ 1 & \hat{m}_{1\to0} & \hat{m}_{1\to1} & \hat{m}_{1\to2} & \hat{m}_{1\to3} & \hat{m}_{1\to4} & \cdots \\ 2 & 0 & \hat{m}_{2\to1} & \hat{m}_{2\to2} & \hat{m}_{2\to3} & \hat{m}_{2\to4} & \cdots \\ 3 & 0 & 0 & \hat{m}_{3\to2} & \hat{m}_{3\to3} & \hat{m}_{3\to4} & \cdots \\ 4 & 0 & 0 & 0 & \hat{m}_{4\to3} & \hat{m}_{4\to4} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{array} . \tag{4.11}$$

For $n \geq 0$, let $\hat{d}_n = P\left\{\hat{L}^2 = n\right\} = \lim_{k\to\infty} P\left\{\hat{L}_k^2 = n\right\}$, i.e., $\hat{L}^2$ is the time-stationary limiting random variable of $\hat{L}_k^2$ and $\hat{d}_n$ is the steady state probability that the number of Class-2 jobs observed by a Class-2 departure is n. Let $G_{\hat{L}^2}(z) = \sum_{n=0}^{\infty} \hat{d}_n z^n$ be the GF of $\hat{L}^2$.

4.4.1 Transition Matrix of the EMC

The distribution of $\hat{L}_{k+1}^2$ given $\hat{L}_k^2$ (i.e., $\hat{m}_{\hat{L}_k^2\to\hat{L}_{k+1}^2}$) is closely related to the SP (Service Pattern, which by (4.1) is only defined when no Class-1 jobs are in the system) in $\hat{D}_k$. The derivation in this section is based on the observation that the SP depends on Class-2 arrivals during $\hat{D}_k$ as follows:

- If $\hat{L}_k^2 \geq 1$: The SP remains μ_2 until the $(k+1)^{st}$ Class-2 departure. The SP is independent of Class-2 arrivals during $\hat{D}_k$.
- If $\hat{L}_k^2 = 0$: The SP is zero until the next Class-2 arrival, and then it becomes μ_1.

The Transition Probabilities for $\hat{L}_k^2 \geq 1$

We know from (4.10) that the transition probabilities of the EMC are determined by $\alpha^{\hat{D}_k}$, which depends on $\hat{D}_k$. Thus, we first derive the LT of $\hat{D}_k$, $LT^{\hat{D}_k}(s)$. Then, using $LT^{\hat{D}_k}(s)$ and (4.2), we express the distribution of $\alpha^{\hat{D}_k}$, and then write the transition probabilities of the EMC using (4.10).

Figure 4.3 illustrates the service process of the $(k+1)^{st}$ Class-2 departure at the MC (not the EMC). At the k^{th} Class-2 departure, the MC enters state $(0, \hat{L}_k^2)$. Because $\hat{L}_k^2 \geq 1$, the

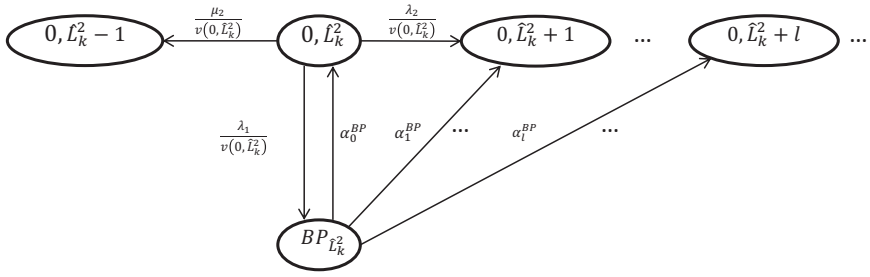

Figure 4.3: MC for the single server case where $\hat{L}_k^2 \geq c = 1$.

rate of exiting from state $(0, \hat{L}_k^2)$ is $v(0, \hat{L}_k^2) = \lambda_1 + \lambda_2 + \mu_2$, thus after an $\exp(\lambda_1 + \lambda_2 + \mu_2)$ distributed time interval, the MC would go to one of the following three states:

- State $BP_{\hat{L}_k^2}$, w.p. $\frac{\lambda_1}{v(0,\hat{L}_k^2)}$. The MC stays in the BP for a time period with a LT of $LT^{BP}(s)$. After this BP, the MC goes to state $(0, \hat{L}_k^2 + l)$ (with $l \geq 0$ the number of Class-2 arrivals during the $BP_{\hat{L}_k^2}$, which can be calculated from (4.6)). Due to the memoryless property and the fact that the SP stays the same, the LT of the time period from when the MC enters $(0, \hat{L}_k^2 + l)$ until the next Class-2 departure is identical to $LT^{\hat{D}_k}(s)$. Therefore, w.p. $\frac{\lambda_1}{v(0,\hat{L}_k^2)}$, $LT^{\hat{D}_k}(s)$ is identical to the LT of the sum of the time until the next event, the length of a BP, and $\hat{D}_k$: $\frac{\lambda_1+\lambda_2+\mu_2}{\lambda_1+\lambda_2+\mu_2+s} LT^{BP}(s) LT^{\hat{D}_k}(s)$.

- State $(0, \hat{L}_k^2 + 1)$, w.p. $\frac{\lambda_2}{v(0,\hat{L}_k^2)}$. Here, using similar reasoning to above: w.p. $\frac{\lambda_2}{v(0,\hat{L}_k^2)}$, $LT^{\hat{D}_k}(s)$ is $\frac{\lambda_1+\lambda_2+\mu_2}{\lambda_1+\lambda_2+\mu_2+s} LT^{\hat{D}_k}(s)$.

- State $(0, \hat{L}_k^2 - 1)$, w.p. $\frac{\mu_2}{v(0,\hat{L}_k^2)}$. The $(k+1)^{st}$ Class-2 departure occurs and here, w.p. $\frac{\mu_2}{v(0,\hat{L}_k^2)}$, $LT^{\hat{D}_k}(s)$ is $\frac{\lambda_1+\lambda_2+\mu_2}{\lambda_1+\lambda_2+\mu_2+s}$.

Using the Total Probability Theorem and multiplying by $(\lambda_1 + \lambda_2 + \mu_2 + s)$, we get

$$(\lambda_1 + \lambda_2 + \mu_2 + s) LT^{\hat{D}_k}(s) = \lambda_1 LT^{BP}(s) LT^{\hat{D}_k}(s) + \lambda_2 LT^{\hat{D}_k}(s) + \mu_2. \tag{4.12}$$

Solving (4.12) gives

$$LT^{\hat{D}_k}(s) = \frac{\mu_2}{\lambda_1 + \mu_2 + s - \lambda_1 LT^{BP}(s)}.$$

We now return to the EMC. To simplify the notation, we let $\alpha_l^{\hat{D}_k} = \frac{(-\lambda_2)^l}{l!} LT^{\hat{D}_k}{}^{(l)}(\lambda_2)$. Then, using (4.2) and (4.10), we get the transition probabilities of the EMC from $\hat{L}_k^2 \geq 1$ to

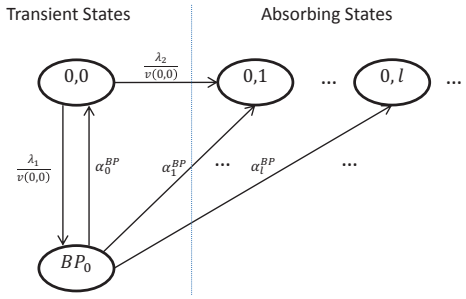

Figure 4.4: MC for the single server case where $\hat{L}_k^2 = 0$.

any $\hat{L}_{k+1}^2 \geq 0$:

$$\hat{m}_{\hat{L}_k^2 \to \hat{L}_{k+1}^2} = \begin{cases} 0 & for\ \hat{L}_{k+1}^2 < \hat{L}_k^2 - 1 \\ \alpha^{\hat{D}_k}_{\hat{L}_{k+1}^2 - \hat{L}_k^2 + 1} & for\ \hat{L}_{k+1}^2 \geq \hat{L}_k^2 - 1 \end{cases} . \tag{4.13}$$

Note that (4.13) characterizes the rows of $\hat{M}$ in (4.11) corresponding to any $i \geq 1$.

The Transition Probabilities for $\hat{L}_k^2 = 0$

If $\hat{L}_k^2 = 0$ when the k^{th} Class-2 departure occurs there are no Class-2 jobs in the system, thus the Class-2 event that happens next must be a Class-2 arrival. The next Class-2 arrival may occur during BP_0, and there may be other Class-2 arrivals during BP_0. Taking this possibility into account, assume that when the service of the next Class-2 arrival is initiated, there are $l \geq 1$ Class-2 jobs in the system, i.e., the MC enters state $(0, l)$ for $l \geq 1$. Note that there are no transitions in the EMC until then.

Due to the memoryless property, the distribution of $\hat{L}_{k+1}^2$ given the MC is in state $(0, l)$ is the same as the distribution of $\hat{L}_{k+1}^2$ given $\hat{L}_k^2 = l$, as given in (4.13) for $l \geq 1$. Thus, to find the one-step transition probabilities of the EMC, we first find the first-passage probability distribution from state $(0, 0)$ to the set of states $\{(0, l) \mid l \geq 1\}$. To do so, we think of the MC after the k^{th} Class-2 departure as a MC with a transient set: $\{(0, 0), BP_0\}$, and an absorbing set: $\{(0, l) \mid l \geq 1\}$. Let $\hat{\Gamma}_{0 \to 0}$ and $\hat{\Gamma}_{0 \to 1^+}$ be the one-step transition matrices from $\{(0, 0), BP_0\}$ to $\{(0, 0), BP_0\}$ and $\{(0, l) \mid l \geq 1\}$, respectively.

In Figure 4.4, we use $v(0, 0) = \lambda_1 + \lambda_2$, depict the arrival process of jobs in $(0, 0)$, and omit details that are not relevant to the development of this case. From Figure 4.4, we can get $\hat{\Gamma}_{0 \to 0}$

and $\hat{\Gamma}_{0\to 1^+}$:

$$\hat{\Gamma}_{0\to 0} = \begin{array}{c} \\ (0,0) \\ BP_0 \end{array} \begin{array}{c} \begin{array}{cc} (0,0) & BP_0 \end{array} \\ \begin{bmatrix} 0 & \frac{\lambda_1}{\lambda_1+\lambda_2} \\ \alpha_0^{BP} & 0 \end{bmatrix} \end{array}, \quad \hat{\Gamma}_{0\to 1^+} = \begin{array}{c} \\ (0,0) \\ BP_0 \end{array} \begin{array}{c} \begin{array}{cccc} (0,1) & (0,2) & (0,3) & \cdots \end{array} \\ \begin{bmatrix} \frac{\lambda_2}{\lambda_1+\lambda_2} & 0 & 0 & \cdots \\ \alpha_1^{BP} & \alpha_2^{BP} & \alpha_3^{BP} & \cdots \end{bmatrix} \end{array}.$$

Let $\hat{\Psi}_{01}$ be the $1 \times \infty$ absorbing distribution matrix from $\{(0,0)\}$ to $\{(0,l) \mid l \geq 1\}$. Using Lemma 2, we can calculate $\hat{\Psi}_{01}$ as:

$$\hat{\Psi}_{01} = [1\ 0]\,(I_{2\times 2} - \hat{\Gamma}_{0\to 0})^{-1}\hat{\Gamma}_{0\to 1^+}. \tag{4.14}$$

Then, we use conditional probability to calculate the transition probabilities for $\hat{L}_k^2 = 0$:

$$\hat{m}_{0\to \hat{L}_{k+1}^2} = \sum_{l=1}^{\hat{L}_{k+1}^2+1} \hat{m}_{l\to \hat{L}_{k+1}^2} P\{(0,l) \mid (0,0)\} \text{ for } \forall \hat{L}_{k+1}^2 \geq 0, \tag{4.15}$$

in which $\hat{m}_{l\to \hat{L}_{k+1}^2}$ is given by (4.13), and $P\{(0,l) \mid (0,0)\}$ is the corresponding probability of absorption in $\{(0,l) \mid l \geq 1\}$ given in (4.14). Note that given the $(k+1)^{st}$ Class-2 departure sees $\hat{L}_{k+1}^2$ Class-2 jobs, l can be at most $\hat{L}_{k+1}^2 + 1$; thus $l \in \left[1, \hat{L}_{k+1}^2 + 1\right]$.

Using (4.15), we can write $\hat{m}_{0\to \hat{L}_{k+1}^2}$ for $\hat{L}_{k+1}^2 \geq 0$ as the product of two matrices:

$$\hat{m}_{0\to \hat{L}_{k+1}^2} = \hat{\Psi}_{01} \begin{bmatrix} \alpha_{\hat{L}_{k+1}^2}^{\hat{D}_k} & \cdots & \alpha_1^{\hat{D}_k} & \alpha_0^{\hat{D}_k} & \mathbf{0}_{1\times\infty} \end{bmatrix}^T. \tag{4.16}$$

Note that (4.16) characterizes the $i = 0$ row of $\hat{M}$ in (4.11). Thus, using (4.13) and (4.16),

we obtain the transition matrix of the EMC in (4.11) as:

$$\hat{M} = \begin{array}{c|c|c|c|c|c|c} & 0 & 1 & 2 & \cdots & n & \cdots \\ \hline 0 & \hat{\Psi}_{01}\begin{bmatrix} \alpha_0^{\hat{D}_k} \\ \mathbf{0}_{\infty\times 1} \end{bmatrix} & \hat{\Psi}_{01}\begin{bmatrix} \alpha_1^{\hat{D}_k} \\ \alpha_0^{\hat{D}_k} \\ \mathbf{0}_{\infty\times 1} \end{bmatrix} & \hat{\Psi}_{01}\begin{bmatrix} \alpha_2^{\hat{D}_k} \\ \alpha_1^{\hat{D}_k} \\ \alpha_0^{\hat{D}_k} \\ \mathbf{0}_{\infty\times 1} \end{bmatrix} & \cdots & \hat{\Psi}_{01}\begin{bmatrix} \alpha_n^{\hat{D}_k} \\ \vdots \\ \alpha_1^{\hat{D}_k} \\ \alpha_0^{\hat{D}_k} \\ \mathbf{0}_{\infty\times 1} \end{bmatrix} & \cdots \\ \hline 1 & \alpha_0^{\hat{D}_k} & \alpha_1^{\hat{D}_k} & \alpha_2^{\hat{D}_k} & \cdots & \alpha_n^{\hat{D}_k} & \cdots \\ \hline 2 & 0 & \alpha_0^{\hat{D}_k} & \alpha_1^{\hat{D}_k} & \cdots & \alpha_{n-1}^{\hat{D}_k} & \cdots \\ \hline 3 & 0 & 0 & \alpha_0^{\hat{D}_k} & \ddots & \alpha_{n-2}^{\hat{D}_k} & \cdots \\ \hline 4 & 0 & 0 & 0 & \vdots & \ddots & \cdots \\ \hline \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{array}. \tag{4.17}$$

4.4.2 Generating Function Approach

In this section, we derive the steady state distribution of the EMC: $\hat{d}_n$, for $n \geq 0$. The equilibrium equations are given by $\left[\hat{d}_0, \hat{d}_1, \ldots\right] \hat{M} = \left[\hat{d}_0, \hat{d}_1, \ldots\right]$. Hence, from (4.17) we get

$$\hat{d}_n = (\left[\hat{d}_1, \hat{d}_2, \ldots\right] + \hat{d}_0 \hat{\Psi}_{01}) \left[\alpha_n^{\hat{D}_k} \ \cdots \ \alpha_1^{\hat{D}_k} \ \alpha_0^{\hat{D}_k} \ \mathbf{0}_{1\times\infty}\right]^T \quad \text{for } \forall n \geq 0. \tag{4.18}$$

Note that (4.18) has an infinite number of unknowns appearing in an infinite (identical) number of equations. To find these unknowns, we calculate the GF, as in the standard $M/G/1$ model (see e.g., Buzacott and Shanthikumar (1993), Section 3.3.2). Multiplying the n^{th} equation in (4.18) by z^n and summing over all n gives

$$G_{\hat{L}^2}(z) = (\left[\hat{d}_1, \hat{d}_2, \ldots\right] + \hat{d}_0 \hat{\Psi}_{01}) \sum_{n=0}^{\infty} \left[\alpha_n^{\hat{D}_k} \ \cdots \ \alpha_1^{\hat{D}_k} \ \alpha_0^{\hat{D}_k} \ \mathbf{0}_{1\times\infty}\right]^T z^n.$$

Let $G_{\alpha^{\hat{D}_k}}(z)$ be the GF of $\alpha^{\hat{D}_k}$ that can be calculated from (4.3) as: $G_{\alpha^{\hat{D}_k}}(z) = LT^{\hat{D}_k}(\lambda_2 -$

$\lambda_2 z$). Then, after some matrix algebra (see Appendix 4.10.1 for details), we get:

$$G_{\hat{L}^2}(z) = -\frac{\hat{d}_0}{(\lambda_1 + \lambda_2 - \alpha_0^{BP}\lambda_1)} \frac{(\lambda_1 + \lambda_2 - z\lambda_2 - \lambda_1 G_{\alpha^{BP}}(z)) G_{\alpha^{\hat{D}_k}}(z)}{z - G_{\alpha^{\hat{D}_k}}(z)}. \tag{4.19}$$

Note that, other than $\hat{d}_0$, all expressions in (4.19) are given in closed form. Therefore, all that is required to express $G_{\hat{L}^2}(z)$ in closed form is a closed-form expression for $\hat{d}_0$, which is derived next.

4.4.3 Finding the Idle Rate: $\hat{d}_0$

To obtain $\hat{d}_0$, we let $z \to 1$ in (4.19) and get (note that $z - G_{\alpha^{\hat{D}_k}}(z)$ is zero when $z \to 1$, so we need to apply L'Hopital's rule to calculate the limit on the right-hand side of (4.19)):

$$1 = -\frac{2\hat{d}_0}{\lambda_1 + \lambda_2 - \mu_1 + \sqrt{(\lambda_1 + \mu_1 + \lambda_2)^2 - 4\lambda_1\mu_1}} \frac{\lambda_2\mu_1\mu_2}{\lambda_1\mu_2 + \lambda_2\mu_1 - \mu_1\mu_2}. \tag{4.20}$$

Solving (4.20) gives us $\hat{d}_0$:

$$\hat{d}_0 = -\frac{\lambda_1\mu_2 + \lambda_2\mu_1 - \mu_1\mu_2}{2\lambda_2\mu_1\mu_2}(\lambda_1 + \lambda_2 - \mu_1 + \sqrt{(\lambda_1 + \lambda_2 + \mu_1)^2 - 4\lambda_1\mu_1}).$$

Substituting $\hat{d}_0$ in (4.19) gives us $G_{\hat{L}^2}(z)$ in closed form:

$$G_{\hat{L}^2}(z) = \frac{2(\lambda_1\mu_2 + \lambda_2\mu_1 - \mu_1\mu_2)}{\mu_2(\lambda_1 + \lambda_2 - \mu_1) + \lambda_2(2\mu_1 - \mu_2)z - \mu_2\sqrt{(\lambda_1 + \lambda_2 + \mu_1 - \lambda_2 z)^2 - 4\lambda_1\mu_1}}.$$

In a single-server queue, the service order in each priority class follows the FIFO rule, so we can use Distributional Little's Law (Bertsimas and Nakazato 1995) to get the LT of Class-2 jobs' response time: $LT^{\hat{R}_2}(s) = G_{\hat{L}^2}(1 - \frac{s}{\lambda_2})$, which, of course, leads to (4.9).

4.5 General case: $c \geq 2$

In this section we return to the multi-server case and follow the same three steps used in Section 4.4 to get the GF of the number of Class-2 jobs in steady state: we give preliminary results in this subsection, establish the transition matrix of the EMC in Section 4.5.1, apply

the Generating Function approach in Section 4.5.2, and derive the constant $\vec{d_0}$ in Section 4.5.3. We provide a summary in Section 4.5.4.

We know from (4.1) that the number of Class-1 jobs in the system affects the service rate of Class-2 jobs. Therefore, we need to track the number of Class-1 jobs seen by a Class-2 departure. Let (L_k^1, L_k^2) denote the state of the EMC; L_k^1 and L_k^2 are the number of Class-1 and Class-2 jobs seen by the k^{th} Class-2 departure, respectively.

To display the infinite dimensional transition matrix of the EMC for $c \geq 2$, we define the ordering of the states as $\{(0,0),\ldots,(c-1,0),\ (0,1),\ldots,(c-1,1),\ldots,(0,n),\ \ldots,(c-1,n),\ldots\}$, i.e., we first list all states with $L_k^2 = 0$, then all states with $L_k^2 = 1$, etc. Let S_n be the set of states with $L_k^2 = n$ in the EMC, i.e., $S_n = \{(0,n),\ldots,(c-1,n)\}$. These states also appear in the MC, when no confusion arises we also use S_n to denote the set of states with $q_2 = n$ in the MC.

Using the ordering of the states defined above, we specify the infinite dimensional transition matrix of the EMC, M. Let the element $m_{(L_k^1,L_k^2)\to(L_{k+1}^1,L_{k+1}^2)}$ in M denote the probability that the $(k+1)^{st}$ Class-2 departure sees (L_{k+1}^1, L_{k+1}^2) given the k^{th} Class-2 departure left behind (L_k^1, L_k^2), i.e., the one-step transition probability of the EMC. Let $M_{i\to j}$ be the $c \times c$ transition matrix in the EMC from S_i to S_j. We illustrate $M_{i\to j}$ here (each row and column is labeled by the corresponding state for easy reference):

$$M_{i\to j} = \begin{array}{c|cccc} & (0,j) & (1,j) & \cdots & (c-1,j) \\ \hline (0,i) & m_{(0,i)\to(0,j)} & m_{(0,i)\to(1,j)} & \cdots & m_{(0,i)\to(c-1,j)} \\ (1,i) & m_{(1,i)\to(0,j)} & m_{(1,i)\to(1,j)} & \cdots & m_{(1,i)\to(c-1,j)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ (c-1,i) & m_{(c-1,i)\to(0,j)} & m_{(c-1,i)\to(1,j)} & \cdots & m_{(c-1,i)\to(c-1,j)} \end{array}. \tag{4.21}$$

Because Class-2 jobs are only served when there are fewer than c Class-1 jobs (i.e., $q_1 < c$) in the system, transitions in the EMC only occur when the MC is in a state (q_1, q_2) with $q_1 < c$. Therefore, the number of Class-1 jobs observed by the k^{th} Class-2 departure must be smaller than c, i.e., $L_k^1 \in [0, c-1]$. Similar to $\hat{D}_k$ and $\alpha^{\hat{D}_k}$ in Section 4.4, we denote D_k as the k^{th} inter-departure time of Class-2 jobs and α^{D_k} as the number of Class-2 arrivals during D_k. Using

the same logic as that used for deriving (4.10), we have

$$L^2_{k+1} = L^2_k - 1 + \alpha^{D_k}. \tag{4.22}$$

The transition matrix, M has the form illustrated in (4.23). Each row and column is labeled by the corresponding set S_i. Every block $M_{i\to j}$ is as illustrated in (4.21). Given (4.22), we have that $M_{i\to j} = 0_{c\times c}$ for $j < i-1$, i.e., all blocks of the lower triangle below the row S_1 in M are zero.

$$M = \begin{array}{c|ccccc|c} & S_0 & S_1 & S_2 & S_3 & S_4 & \cdots \\ \hline S_0 & M_{0\to 0} & M_{0\to 1} & M_{0\to 2} & M_{0\to 3} & M_{0\to 4} & \cdots \\ S_1 & M_{1\to 0} & M_{1\to 1} & M_{1\to 2} & M_{1\to 3} & M_{1\to 4} & \cdots \\ S_2 & 0_{c\times c} & M_{2\to 1} & M_{2\to 2} & M_{2\to 3} & M_{2\to 4} & \cdots \\ S_3 & 0_{c\times c} & 0_{c\times c} & M_{3\to 2} & M_{3\to 3} & M_{3\to 4} & \cdots \\ S_4 & 0_{c\times c} & 0_{c\times c} & 0_{c\times c} & M_{4\to 3} & M_{4\to 4} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{array}. \tag{4.23}$$

For $i = 0, \ldots, c-1$ and $n \geq 0$, let $d_{in} = P\left\{(L^1, L^2) = (i, n)\right\} = \lim_{k\to\infty} P\left\{(L^1_k, L^2_k) = (i, n)\right\}$, so that (L^1, L^2) is the time-stationary limiting random variable of (L^1_k, L^2_k), and d_{in} is the steady state probability that the number of Class-1 and Class-2 jobs observed by a Class-2 departure is i and n, respectively.

Let $\vec{d_n} = (d_{0n}, \ldots, d_{(c-1)n})$, i.e., $\vec{d_n}$ is the $1 \times c$ row vector of steady state probabilities that the EMC is in S_n. Let $\vec{d} = \left[\vec{d_0}\ \vec{d_1}\ \vec{d_2}\ \cdots\right]$, i.e., $\vec{d}$ is the $1 \times \infty$ row vector composed of an infinite number of row vectors, $\vec{d_n}$ for $n \geq 0$. Let $G_{(i,L^2)}(z) = \sum_{n=0}^{\infty} d_{in} z^n$ be the GF of L^2 when $L^1 = i$, i.e., of the joint event $L^2 = n$ and $L^1 = i$ (*not* of the event $L^2 = n$ *given* $L^1 = i$), for $i \in [0, c-1]$. So $\sum_{n=0}^{\infty} \vec{d_n} z^n = \left[G_{(0,L^2)}(z), \ldots, G_{(c-1,L^2)}(z)\right]$ is the $1 \times c$ row vector of GF of L^2 for $L^1 \in [0, c-1]$.

Because a Class-2 departure can only see $0, \ldots, c-1$ Class-1 jobs, once we get $G_{(i,L^2)}(z)$, using the total probability theorem, we have the GF of the number of Class-2 jobs at Class-2 departures:

$$G_{L^2}(z) = \sum_{i=0}^{c-1} G_{(i,L^2)}(z). \tag{4.24}$$

As in Section 4.4.1, we derive the transition matrix of the EMC based on the observation that the SP depends on Class-2 arrivals in D_k as follows:

- If $L_k^2 \geq c$: The SP remains $SP(c)$ at least until the $(k+1)^{st}$ Class-2 departure, independent of Class-2 arrivals during D_k.
- If $L_k^2 \in [1, c-1]$: If no Class-2 arrivals happen before the $(k+1)^{st}$ Class-2 departure, the SP remains $SP(L_k^2)$ until the $(k+1)^{st}$ Class-2 departure. Otherwise, the SP is $SP(L_k^2)$ until the next Class-2 arrival, and then the SP becomes $SP(L_k^2+1)$. (As the next Class-2 arrival may happen during $BP_{L_k^2}$ together with other l Class-2 arrivals, when the MC leaves $BP_{L_k^2}$, the SP would be $SP(L_k^2+l+1)$, $l \geq 0$.)
- If $L_k^2 = 0$: The SP remains $SP(0)$ until the next Class-2 arrival, and then the SP becomes $SP(1)$ (or $SP(l+1)$, $l \geq 0$, see the discussion in previous bullet point.)

For simplicity, we next demonstrate the use of the ideas above to derive M, for the special case of $c = 2$. The general case with $c > 2$ can be analyzed in similarly. However, $c = 2$ is the only case for which we derive a closed-form expression for the number of Class-2 jobs in steady state.

4.5.1 Transition Matrix of the EMC

In Section 4.4.1 we derived the LT of $\hat{D}_k$, $LT^{\hat{D}_k}$, expressed the distribution of $\alpha^{\hat{D}_k}$ using (4.2), and then wrote the transition probabilities of the EMC, at the moment of the $(k+1)^{st}$ Class-2 departure using (4.22). We follow the same process here, for $c = 2$. We first derive LT^{D_k} while considering the SP when $L_k^2 \geq 2$, and then $L_k^2 = 1$, and finally $L_k^2 = 0$.

The Transition Probabilities for $L_k^2 \geq 2$

Notice that, because r_{q_1,q_2} depends on the number of Class-1 jobs in the network, D_k depends on the values of L_k^1 and L_{k+1}^1. For every $L_k^2 \geq 2$, there are four feasible combinations of L_k^1 and L_{k+1}^1: $0 \to 0$, $0 \to 1$, $1 \to 0$ and $1 \to 1$. In contrast to the single-server case where we had one possible inter-departure time distribution, here we have 2^2 different inter-departure time

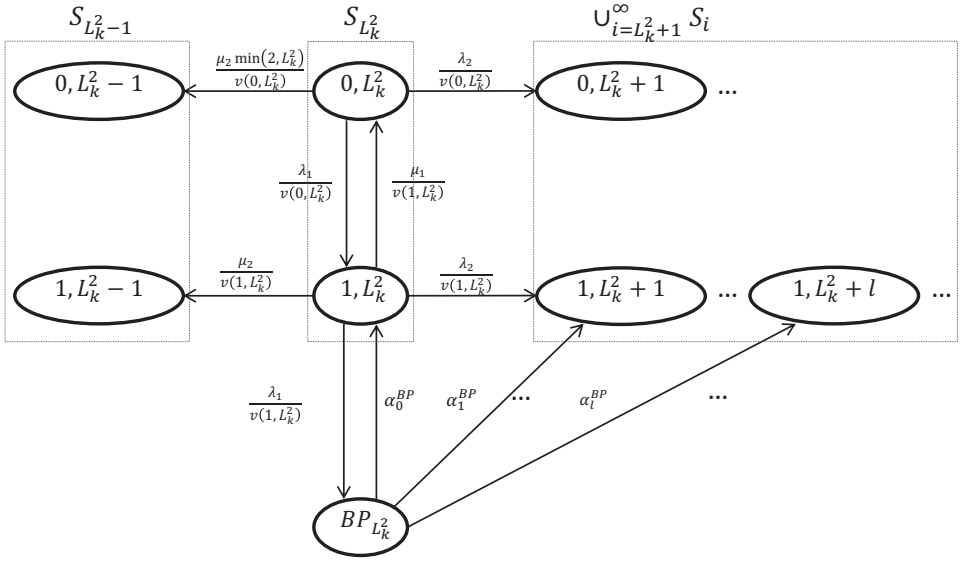

Figure 4.5: MC for the $c = 2$ servers case where $L_k^2 \geq 1$.

distributions in the EMC. (For general $c > 2$ we have c^2 different inter-departure times when $L_k^2 \geq c$.)

Let $LT^{L_k^1, L_{k+1}^1}(s)$ be the LT of D_k conditioning on L_k^1 and L_{k+1}^1, given $L_k^2 \geq 2$ (we omit the latter dependency for notational convenience). For example, $LT^{00}(s)$ is the LT of D_k when the k^{th} and $(k+1)^{st}$ Class-2 departures see no Class-1 jobs in the network at their departures.

Figure 4.5 illustrates the service and arrival process of the Class-2 jobs in the MC after the k^{th} Class-2 departure where $L_k^2 \geq 1$, omitting details that are not relevant.

We next discuss the possible steps of the MC after the k^{th} Class-2 departure to express $LT^{00}(s)$, $LT^{01}(s)$, $LT^{10}(s)$, and $LT^{11}(s)$. Consider $LT^{10}(s)$ for example. The rate of exiting from any state $(1, L_k^2)$ is $v(1, L_k^2) = \lambda_1 + \lambda_2 + \mu_1 + \mu_2$, thus after an $\exp(\lambda_1 + \lambda_2 + \mu_1 + \mu_2)$ distributed time interval, the MC would move to one of the following four states:

- State $BP_{L_k^2}$, w.p. $\frac{\lambda_1}{v(1, L_k^2)}$. Similar reasoning as in Section 4.4.1 gives that w.p. $\frac{\lambda_1}{v(1, L_k^2)}$, $LT^{10}(s)$ is $\frac{\lambda_1+\lambda_2+\mu_1+\mu_2}{\lambda_1+\lambda_2+\mu_1+\mu_2+s} LT^{BP}(s) LT^{10}(s)$.

- State $(1, L_k^2 + 1)$, w.p. $\frac{\lambda_2}{v(1, L_k^2)}$. Similar reasoning gives that w.p. $\frac{\lambda_2}{v(1, L_k^2)}$, $LT^{10}(s)$ is $\frac{\lambda_1+\lambda_2+\mu_1+\mu_2}{\lambda_1+\lambda_2+\mu_1+\mu_2+s} LT^{10}(s)$.

- State $(0, L_k^2)$, w.p. $\frac{\mu_1}{v(1, L_k^2)}$. From the memoryless property, the LT of the time from when the MC enters state $(0, L_k^2)$ until the next Class-2 departure occurs (with $L_{k+1}^1 = 0$) is

$LT^{00}(s)$. Thus, w.p. $\frac{\mu_1}{v(1,L_k^2)}$, $LT^{10}(s)$ is $\frac{\lambda_1+\lambda_2+\mu_1+\mu_2}{\lambda_1+\lambda_2+\mu_1+\mu_2+s}LT^{00}(s)$.

- State $(1, L_k^2-1)$, w.p. $\frac{\mu_2}{v(1,L_k^2)}$. Here, the next Class-2 departure occurs, but L_{k+1}^1 is not 0, so in this case, a transition in the EMC from $L_k^1 = 1$ to $L_{k+1}^1 = 0$ is infeasible. Therefore, w.p. $\frac{\mu_2}{v(1,L_k^2)}$, $LT^{10}(s)$ is 0.

Using the Total Probability Theorem and multiplying by $\lambda_1 + \lambda_2 + \mu_1 + \mu_2 + s$, we get

$$(\lambda_1 + \lambda_2 + \mu_1 + \mu_2 + s)LT^{10}(s) = \lambda_1 LT^{BP}(s)LT^{10}(s) + \lambda_2 LT^{10}(s) + \mu_1 LT^{00}(s). \quad (4.25)$$

Using similar logic, we derive the following three additional equations:

$$\begin{aligned}
(\lambda_1 + \lambda_2 + 2\mu_2 + s)LT^{00}(s) &= \lambda_1 LT^{10}(s) + \lambda_2 LT^{00}(s) + 2\mu_2; & (4.26)\\
(\lambda_1 + \lambda_2 + 2\mu_2 + s)LT^{01}(s) &= \lambda_1 LT^{11}(s) + \lambda_2 LT^{01}(s); & (4.27)\\
(\lambda_1 + \lambda_2 + \mu_1 + \mu_2 + s)LT^{11}(s) &= \lambda_1 LT^{BP}(s)LT^{11}(s) + \lambda_2 LT^{11}(s) + \mu_1 LT^{01}(s) + \mu_2. & (4.28)
\end{aligned}$$

Thus, $(4.25-4.28)$ give four equations with four unknowns, which can be solved in closed form. Using $\Theta = ((\lambda_1 + 2\mu_2 + s)(\lambda_1 + \mu_1 + \mu_2 + s - \lambda_1 LT^{BP}(s)) - \lambda_1\mu_1)^{-1}$, we get:

$$\begin{aligned}
LT^{00}(s) &= 2\mu_2(\lambda_1 + \mu_1 + \mu_2 + s - \lambda_1 LT^{BP}(s))\Theta; \qquad LT^{01}(s) = \lambda_1\mu_2\Theta;\\
LT^{11}(s) &= \mu_2(\lambda_1 + 2\mu_2 + s)\Theta; \qquad\qquad\qquad\qquad\quad LT^{10}(s) = 2\mu_1\mu_2\Theta.
\end{aligned}$$

We let $\alpha_l^{L_k^1,L_{k+1}^1} = \frac{(-\lambda_2)^l}{l!}LT^{L_k^1,L_{k+1}^1\,(l)}(\lambda_2)$, then using (4.2) and (4.22), we get, for $L_k^2 \geq 2$, the transition probabilities of the EMC:

$$m_{(L_k^1,L_k^2)\to(L_{k+1}^1,L_{k+1}^2)} = \begin{cases} 0 \text{ for } L_{k+1}^2 < L_k^2 - 1 \\ \alpha_{L_{k+1}^2-L_k^2+1}^{L_k^1,L_{k+1}^1} \text{ for } L_{k+1}^2 \geq L_k^2 - 1 \end{cases}. \quad (4.29)$$

Letting $\mathscr{A}_l = \begin{bmatrix} \alpha_l^{00} & \alpha_l^{01} \\ \alpha_l^{10} & \alpha_l^{11} \end{bmatrix}$ be the 2×2 matrix of the probability that $\alpha^{D_k} = l$, as a function

of the four different D_k, we get the matrices $M_{L_k^2 \to L_{k+1}^2}$ for $L_k^2 \geq 2$ and $L_{k+1}^2 \geq 0$:

$$M_{L_k^2 \to L_{k+1}^2} = \begin{cases} 0_{2\times 2} \text{ if } L_{k+1}^2 < L_k^2 - 1 \\ \mathscr{A}_{L_{k+1}^2 - L_k^2 + 1} \text{ if } L_{k+1}^2 \geq L_k^2 - 1 \end{cases}. \tag{4.30}$$

Note that (4.30) characterizes the rows of M in (4.23) that correspond to any S_i with $i \geq 2$.

The Transition Probabilities for $L_k^2 = 1$

Here, since we assume $L_k^2 = 1$, when the k^{th} Class-2 departure occurs, the MC moved into S_1. Before the next Class-2 arrival or departure, there may be many Class-1 arrivals and departures, so the MC may move among states in $S_1 \cup BP_1$. When the MC leaves $S_1 \cup BP_1$, it may move to S_0 (Class-2 departure occurs first) or to $\cup_{i=2}^{\infty} S_i$ (Class-2 arrival occurs first). In both of these cases we can establish the conditional distribution of (L_{k+1}^1, L_{k+1}^2). Thus, finding the one-step transition probabilities of the EMC is reduced to finding the first-passage probability distribution from S_1 to S_0 and $\cup_{i=2}^{\infty} S_i$.

We again think of the MC after the k^{th} Class-2 departure as a MC with a transient set: $S_1 \cup BP_1$, and absorbing sets: $\cup_{i=2}^{\infty} S_i \cup S_0$. In the MC, let $\Gamma_{1\to 1}$, $\Gamma_{1\to 0}$ and $\Gamma_{1\to 2^+}$ be the one-step transition matrices from $S_1 \cup BP_1$ to $S_1 \cup BP_1$, S_0, and $\cup_{i=2}^{\infty} S_i$, respectively.

From Figure 4.5, we can see that $\Gamma_{1\to 1}$, $\Gamma_{1\to 0}$ and $\Gamma_{1\to 2^+}$ are:

$$\Gamma_{1\to 1} = \begin{array}{c} \\ (0,1) \\ (1,1) \\ BP_1 \end{array} \begin{array}{c} \begin{array}{ccc} (0,1) & (1,1) & BP_1 \end{array} \\ \begin{bmatrix} 0 & \frac{\lambda_1}{v(0,1)} & 0 \\ \frac{\mu_1}{v(1,1)} & 0 & \frac{\lambda_1}{v(1,1)} \\ 0 & \alpha_0^{BP} & 0 \end{bmatrix} \end{array}, \quad \Gamma_{1\to 0} = \begin{array}{c} \\ (0,1) \\ (1,1) \\ BP_1 \end{array} \begin{array}{c} \begin{array}{cc} (0,0) & (1,0) \end{array} \\ \begin{bmatrix} \frac{\mu_2}{v(0,1)} & 0 \\ 0 & \frac{\mu_2}{v(1,1)} \\ 0 & 0 \end{bmatrix} \end{array},$$

$$\text{and } \Gamma_{1\to 2^+} = \begin{array}{c} \\ (0,1) \\ (1,1) \\ BP_1 \end{array} \begin{array}{c} \begin{array}{ccccc} (0,2) & (1,2) & (0,3) & (1,3) & \cdots \end{array} \\ \begin{bmatrix} \frac{\lambda_2}{v(0,1)} & 0 & 0 & 0 & \cdots \\ 0 & \frac{\lambda_2}{v(1,1)} & 0 & 0 & \cdots \\ 0 & \alpha_1^{BP} & 0 & \alpha_2^{BP} & \cdots \end{bmatrix} \end{array}.$$

To derive the one-step transition probabilities of the EMC, we next discuss the possible steps of the MC, when it leaves the set $S_1 \cup BP_1$.

- If the MC moves to S_0, then the $(k+1)^{st}$ Class-2 departure happens before the next Class-2 arrival. Using Lemma 2 and an algebra software such as MAPLE, the probability of absorption in S_0 (starting at S_1) is

$$\Psi_{10} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \cdot (I_{3\times 3} - \Gamma_{1\to 1})^{-1}\Gamma_{1\to 0} \tag{4.31}$$

$$= \frac{\mu_2 \begin{bmatrix} \lambda_1+\lambda_2+\mu_1+\mu_2-\alpha_0^{BP}\lambda_1 & \lambda_1 \\ \mu_1 & \lambda_1+\lambda_2+\mu_2 \end{bmatrix}}{(\lambda_1+\lambda_2+\mu_2-\alpha_0^{BP}\lambda_1)(\lambda_1+\lambda_2+\mu_2)+\lambda_2\mu_1+\mu_1\mu_2}. \tag{4.32}$$

  At this absorption time the EMC moves into a state $(L_{k+1}^1, L_{k+1}^2) \in S_0$. Thus, the transition matrix from S_1 to S_0 in the EMC, $M_{1\to 0}$ is Ψ_{10}.

- If the MC moves to $\cup_{i=2}^{\infty} S_i$, then a Class-2 arrival happens before the $(k+1)^{st}$ Class-2 departure. (Note that this Class-2 arrival may have occurred during the BP_1 and it may not be the only Class-2 arrival during the BP_1; the number of Class-2 arrivals during the BP_1 can be calculated from (4.6).) From Lemma 2, we can calculate the absorbing distribution matrix from S_1 to $\cup_{i=2}^{\infty} S_i$:

$$\Psi_{12} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \cdot (I_{3\times 3} - \Gamma_{1\to 1})^{-1}\Gamma_{1\to 2^+}. \tag{4.33}$$

  After the MC enters states within $\cup_{i=2}^{\infty} S_i$, there are more than one Class-2 jobs in the system and the SP is identical to the one for $L_k^2 \geq 2$. Using the memoryless property, the distribution of (L_{k+1}^1, L_{k+1}^2) given the MC is in $\cup_{i=2}^{\infty} S_i$ is identical to the distribution of (L_{k+1}^1, L_{k+1}^2) given $(L_k^1, L_k^2) \in \cup_{i=2}^{\infty} S_i$, (4.29). Then, we use conditional probability to calculate transition probabilities of the EMC:

$$m_{(L_k^1,1)\to(L_{k+1}^1,L_{k+1}^2)} = \sum_{(q_1,q_2)\in\cup_{i=2}^{L_{k+1}^2+1} S_i} m_{(q_1,q_2)\to(L_{k+1}^1,L_{k+1}^2)} P\left\{(q_1,q_2) \mid (L_k^1,1)\right\}, \tag{4.34}$$

  in which $m_{(q_1,q_2)\to(L_{k+1}^1,L_{k+1}^2)}$ is given in (4.29) and $P\left\{(q_1,q_2) \mid (L_k^1,1)\right\}$ is the correspond-

ing probability of absorption in $\cup_{i=2}^{\infty} S_i$ given in (4.33). The upper bound of q_2 is L_{k+1}^2+1, because for the $(k+1)^{st}$ Class-2 departure to see L_{k+1}^2 Class-2 jobs, q_2 can be at most L_{k+1}^2+1. The lower bound of q_2 is 2, because (q_1, q_2) is in $\cup_{i=2}^{\infty} S_i$.

From (4.32) and (4.34), we get matrices $M_{1\to L_{k+1}^2}$ for $L_{k+1}^2 \geq 0$, expressing the S_1 row of M in (4.23)

$$M_{1\to L_{k+1}^2} = \begin{cases} \Psi_{10} & \text{for } L_{k+1}^2 = 0 \\ \Psi_{12}\begin{bmatrix} \mathscr{A}_{L_{k+1}^2-1}^T & \cdots & \mathscr{A}_1^T & \mathscr{A}_0^T & \mathbf{0}_{2\times\infty} \end{bmatrix}^T & \text{for } L_{k+1}^2 \geq 1 \end{cases}. \tag{4.35}$$

The Transition Probabilities for $L_k^2 = 0$

Using a similar analysis as in Section 4.5.1, we obtain the matrices $M_{0\to L_{k+1}^2}$ for $L_{k+1}^2 \geq 0$, characterizing the S_0 row of M in (4.23) (the detailed procedure is given in Appendix 4.10.2):

$$M_{0\to L_{k+1}^2} = \begin{cases} \Psi_{01}\Psi_{10} & \text{for } L_{k+1}^2 = 0 \\ (\Psi_{01}\Psi_{12}+\Psi_{02})\begin{bmatrix} \mathscr{A}_{n-1}^T & \cdots & \mathscr{A}_1^T & \mathscr{A}_0^T & \mathbf{0}_{2\times\infty} \end{bmatrix}^T & \text{for } L_{k+1}^2 \geq 1 \end{cases}. \tag{4.36}$$

Thus, using (4.30), (4.35) and (4.36), we obtain the transition matrix of the EMC in (4.23) as:

$$M = \begin{array}{c|c|c|c|c} & S_0 & S_1 & \cdots & S_n & \cdots \\ \hline S_0 & \Psi_{01}\Psi_{10} & (\Psi_{01}\Psi_{12}+\Psi_{02}) \begin{bmatrix} \mathscr{A}_0 \\ \mathbf{0}_{\infty\times 2} \end{bmatrix} & \cdots & (\Psi_{01}\Psi_{12}+\Psi_{02}) \begin{bmatrix} \mathscr{A}_{n-1} \\ \vdots \\ \mathscr{A}_1 \\ \mathscr{A}_0 \\ \mathbf{0}_{\infty\times 2} \end{bmatrix} & \cdots \\ \hline S_1 & \Psi_{10} & \Psi_{12} \begin{bmatrix} \mathscr{A}_0 \\ \mathbf{0}_{\infty\times 2} \end{bmatrix} & \cdots & \Psi_{12} \begin{bmatrix} \mathscr{A}_{n-1} \\ \vdots \\ \mathscr{A}_1 \\ \mathscr{A}_0 \\ \mathbf{0}_{\infty\times 2} \end{bmatrix} & \cdots \\ \hline S_2 & 0 & \mathscr{A}_0 & \cdots & \mathscr{A}_{n-1} & \cdots \\ \hline S_3 & 0 & 0 & \ddots & \mathscr{A}_{n-2} & \cdots \\ \hline S_4 & 0 & 0 & \vdots & \ddots & \cdots \\ \hline \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{array} . \tag{4.37}$$

4.5.2 Generating Function Approach

We now derive the steady state distribution of the EMC for the case of $c = 2$. Recalling that $\vec{d}$ is the row vector of the steady state distribution of the EMC, the equilibrium equations are given by $\vec{d} \cdot M = \vec{d}$, so from (4.37)

$$\vec{d_n} = \begin{cases} (\vec{d_1} + \vec{d_0}\Psi_{01})\Psi_{10} & if\ n = 0 \\ (\left[\vec{d_2}, \vec{d_3}, ...\right] + \vec{d_1}\Psi_{12} + \vec{d_0}(\Psi_{01}\Psi_{12} + \Psi_{02})) \begin{bmatrix} \mathscr{A}_{n-1}^T & \cdots & \mathscr{A}_1^T & \mathscr{A}_0^T & \mathbf{0}_{2\times\infty} \end{bmatrix}^T & if\ n \geq 1 \end{cases} . \tag{4.38}$$

Note that (4.38), just as (4.18), has an infinite number of unknowns appearing in an infinite (identical) number of equations. To find these unknowns, we calculate the GF as in the standard

$M/G/1$ queue. Multiplying the n^{th} equation in (4.38) by z^n and summing over all n:

$$\begin{aligned} & [G_{(0,L^2)}(z), G_{(1,L^2)}(z)] \\ = \; & \vec{d_0} + (\left[\vec{d_2}, \vec{d_3}, \ldots\right] + \vec{d_1}\Psi_{12} + \vec{d_0}(\Psi_{01}\Psi_{12} + \Psi_{02})) \sum_{n=1}^{\infty} \left[\mathscr{A}_{n-1}^T \; \cdots \; \mathscr{A}_1^T \; \mathscr{A}_0^T \; \mathbf{0}_{2\times\infty}\right]^T z^n. \end{aligned}$$

With some matrix calculations (see Appendix 4.10.1 for details), we get:

$$\left[G_{(0,L^2)}(z), G_{(1,L^2)}(z)\right] = \vec{d_0}\mathscr{D}(z), \tag{4.39}$$

where $\mathscr{D}(z)$ is given in closed form in Appendix 4.10.1.

Therefore, if we can express $\vec{d_0}$ in closed form as well, we could use (4.39) to express $\left[G_{(0,L^2)}(z), G_{(1,L^2)}(z)\right]$ in closed form. Then, we get the GF of L^2:

$$G_{L^2}(z) = G_{(0,L^2)}(z) + G_{(1,L^2)}(z). \tag{4.40}$$

If we further assume that the service order in each priority class follows the FIFO rule, we can use the Distributional Little's Law (Bertsimas and Nakazato 1995) to get the LT of Class-2 jobs' response time:

$$LT^{R_2}(s) = G_{(0,L^2)}(1 - \frac{s}{\lambda_2}) + G_{(1,L^2)}(1 - \frac{s}{\lambda_2}).$$

The next section is devoted to deriving $\vec{d_0}$.

4.5.3 Expressing $\vec{d_0}$ in Closed Form

To obtain $\vec{d_0}$, we let $z \to 1$ in (4.39) and get

$$\left[G_{(0,L^2)}(1), G_{(1,L^2)}(1)\right] = \vec{d_0} \cdot \lim_{z\to 1} \mathscr{D}(z). \tag{4.41}$$

Notice that the denominator of $\mathscr{D}(z)$ is zero when $z \to 1$, so we need to apply L'Hopital's rule to calculate $\lim_{z\to 1} \mathscr{D}(z)$. The value of $\lim_{z\to 1} \mathscr{D}(z)$ is determined by $G_{\alpha^{BP}}(z), G_{\alpha^{00}}(z), G_{\alpha^{11}}(z), G_{\alpha^{01}}(z), G_{\alpha}$ and their first order derivatives, which can all be calculated from (4.7) and (4.62).

Note that (4.41) is composed of two equations with four unknowns: $G_{(0,L^2)}(1)$, $G_{(1,L^2)}(1)$,

d_{00} and d_{10}. Another equation is the normalization requirement

$$G_{(0,L^2)}(1) + G_{(1,L^2)}(1) = 1. \tag{4.42}$$

Thus, to find a closed-form expression of $\left[G_{(0,L^2)}(z), G_{(1,L^2)}(z)\right]$, we need another linearly independent equation of these four variables. To find this equation, we focus on the value of $\varphi_1 = \frac{d_{10}}{d_{10}+d_{00}}$.

Let a *Level-j Class-2 busy period* ($j = 0, 1, \ldots$) start once a Class-2 job arrives at the system when j Class-2 jobs are present (but not necessarily in service), and terminate at the first time the number of Class-2 jobs in the system drops to j. Let "a Level-j Class-2 busy period starts with i Class-1 jobs" denote that the first Class-2 arrival in this Level-j Class-2 busy period sees i Class-1 jobs, similarly "a Level-j Class-2 busy period ends with i Class-1 jobs" denote that the Class-2 departure that ends this Level-j Class-2 busy period sees i Class-1 jobs. Recall that, in our M/M/2 queue, a Class-2 departure sees either zero or one Class-1 job. With these definitions, φ_1 is the probability that a Level-0 Class-2 busy period ends with one Class-1 job.

Let Π_i be the probability that a Level-0 Class-2 busy period starts with $i \geq 0$ Class-1 jobs. Let F_i be the probability that a Level-0 Class-2 busy period that started with $i \geq 0$ Class-1 jobs ends with one Class-1 job. Note that, in the $c = 2$ case, the probability that a Level-j Class-2 busy period ($j = 1, 2, \ldots$) that started with a fixed $i \geq 0$ Class-1 jobs ends with one Class-1 jobs is the same for any Level-j Class-2 busy period for any $j = 1, 2, \ldots$. Let B_i be this probability.

Using the Total Probability Theorem, we have

$$\varphi_1 = \sum_{i=0}^{\infty} \Pi_i F_i \ . \tag{4.43}$$

Thus, if we can find F_i and Π_i in closed form, we can also express φ_1 in closed form.

We then discuss the next possible events, and use the memoryless property to write recursive expressions for F_i and B_i. For example, if a Level-0 Class-2 busy period starts with no Class-1 jobs (i.e., a Class-2 job arrives at an empty system), then three events may happen in the system:

1. Class-1 arrival, w.p. $\frac{\lambda_1}{\lambda_1+\lambda_2+\mu_2}$. Thus, one Class-1 job is in the system. Then, due to the memoryless property, F_0 is identical to F_1, the probability that a Level-0 Class-2 busy period that started with one Class-1 job ends with one Class-1 job.

2. Class-2 arrival, w.p. $\frac{\lambda_2}{\lambda_1+\lambda_2+\mu_2}$. A Level-1 Class-2 busy period is started. It ends with one Class-2 job and either zero or one Class-1 job:

   (a) One Class-1 job, w.p. B_0. Then, due to the memoryless property, a Level-0 Class-2 busy period starts with one Class-1 job, and it will end with one Class-1 job w.p. F_1.

   (b) No Class-1 jobs, w.p. $1 - B_0$. Then, due to the memoryless property, a Level-0 Class-2 busy period starts with no Class-1 jobs, and it will end with one Class-1 job w.p. F_0.

3. Class-2 departure, w.p. $\frac{\mu_2}{\lambda_1+\lambda_2+\mu_2}$. A Level-0 Class-2 busy period ends with no Class-1 jobs. That is, it ends with one Class-1 job w.p. 0.

Using the Total Probability Theorem and multiplying by $\lambda_1 + \lambda_2 + \mu_2$, we get

$$(\lambda_1 + \lambda_2 + \mu_2)F_0 = \lambda_1 F_1 + \lambda_2(B_0 F_1 + (1 - B_0)F_0) + \mu_2 \cdot 0. \tag{4.44}$$

Similar logic yields

$$\begin{aligned}(\lambda_1 + \lambda_2 + \mu_1 + \mu_2)F_1 &= \lambda_1 F_2 + \lambda_2(B_1 F_1 + (1 - B_1)F_0) + \mu_1 F_0 + \mu_2, &(4.45)\\ (\lambda_1 + \lambda_2 + 2\mu_1)F_i &= \lambda_1 F_{i+1} + \lambda_2(B_i F_1 + (1 - B_i)F_0) + 2\mu_1 F_{i-1} \text{ for } i \geq 2, &(4.46)\end{aligned}$$

for a Level-0 Class-2 busy period; and

$$\begin{aligned}(\lambda_1 + \lambda_2 + 2\mu_2)B_0 &= \lambda_1 B_1 + \lambda_2(B_0 B_1 + (1 - B_0)B_0) + 2\mu_2 \cdot 0, &(4.47)\\ (\lambda_1 + \lambda_2 + \mu_1 + \mu_2)B_1 &= \lambda_1 B_2 + \lambda_2(B_1 B_1 + (1 - B_1)B_0) + \mu_1 B_0 + \mu_2, &(4.48)\\ (\lambda_1 + \lambda_2 + 2\mu_1)B_i &= \lambda_1 B_{i+1} + \lambda_2(B_i B_1 + (1 - B_i)B_0) + 2\mu_1 B_{i-1} \text{ for } i \geq 2, &(4.49)\end{aligned}$$

for Level-j Class-2 busy periods, $j = 1, 2, \ldots$.

Note that B_i is independent of F_i, but F_i depends on B_i. Therefore, we first express B_i.

Lemma 3 *B_i is given by*

$$B_i = \begin{cases} \frac{\lambda_1 \Delta_0^B}{2\mu_2 - \lambda_2 \Delta_0^B} & \text{for } i = 0 \\ \frac{\lambda_1 \Delta_0^B}{2\mu_2 - \lambda_2 \Delta_0^B} + \Delta_0^B + \kappa \frac{g - g^i}{1-g} & \text{for } i \geq 1 \end{cases},$$

where $\Delta_0^B = \frac{-2\mu_1 + g\lambda_1 + g\lambda_2 + 2g\mu_1 - g^2\lambda_1}{g\lambda_2}$, $\kappa = \frac{1}{\lambda_1 g}((\lambda_1 + \lambda_2 + \mu_1 + \mu_2 - \lambda_2\Delta_0^B)\Delta_0^B - \frac{\lambda_1\mu_2\Delta_0^B}{2\mu_2 - \lambda_2\Delta_0^B} - \mu_2)$, *and g is the only root in $(0,1)$ of the following quartic function:*

$$\lambda_1^2 g^4 + \lambda_1(2\mu_2 - \lambda_1 - \lambda_2 - 4\mu_1)g^3 + 2(\mu_1(2\mu_1 + 4\lambda_1 + \lambda_2 - 2\mu_2) - \lambda_1\mu_2)g^2 + 4\mu_1(\mu_2 - \lambda_1 - \lambda_2 - 3\mu_1)g + 8\mu_1^2.$$

Then, using the same technique, we can express F_i.

Lemma 4 *F_i is given by*

$$F_i = \begin{cases} \frac{2\lambda_1\mu_2\Delta_0^F}{\mu_2(2\mu_2 - \lambda_2\Delta_0^B)} & \text{for } i = 0 \\ \frac{2\lambda_1 + 2\mu_2 - \lambda_2\Delta_0^B}{2\mu_2 - \lambda_2\Delta_0^B}\Delta_0^F + \xi_1\frac{h - h^i}{1-h} + \xi_2\frac{g - g^i}{1-g} & \text{for } i \geq 1 \end{cases}, \tag{4.50}$$

where $h = \frac{1}{2\lambda_1}((\lambda_1 + \lambda_2 + 2\mu_1) - \sqrt{(\lambda_1 + \lambda_2 + 2\mu_1)^2 - 8\lambda_1\mu_1})$, *and*

$$\begin{bmatrix} \xi_1 \\ \xi_2 \\ \Delta_0^F \end{bmatrix} = H^{-1} \begin{bmatrix} -\frac{\mu_2}{\lambda_1} \\ \frac{1}{\lambda_1}(\mu_2 - \frac{1}{\lambda_1}\mu_2(\lambda_1 + \lambda_2 + 2\mu_1)) \\ (\frac{2}{\lambda_1^2}\mu_1\mu_2 + \frac{1}{\lambda_1^2}(\mu_2 - \frac{1}{\lambda_1}\mu_2(\lambda_1 + \lambda_2 + 2\mu_1))(\lambda_1 + \lambda_2 + 2\mu_1)) \end{bmatrix},$$

in which

$$H = \begin{bmatrix} h & g & -\frac{\lambda_1 + \lambda_2 + \mu_1 + \mu_2 - \lambda_2\Delta_0^B}{\lambda_1} \\ h^2 & g^2 & \frac{\mu_1 + \mu_2 + g\kappa\lambda_2}{\lambda_1} - \frac{\lambda_1 + \lambda_2 + 2\mu_1}{\lambda_1^2}(\lambda_1 + \lambda_2 + \mu_1 + \mu_2 - \lambda_2\Delta_0^B) + \frac{2\mu_2}{2\mu_2 - \lambda_2\Delta_0^B} \\ h^3 & g^3 & \frac{2\mu_1}{\lambda_1^2}(\lambda_1 + \lambda_2 + \mu_1 + \mu_2 - \lambda_2\Delta_0^B) + \frac{\lambda_2\kappa g^2}{\lambda_1} \\ & & + \frac{\lambda_1 + \lambda_2 + 2\mu_1}{\lambda_1^2}(\mu_1 + \mu_2 + g\kappa\lambda_2 - \frac{\lambda_1 + \lambda_2 + 2\mu_1}{\lambda_1}(\lambda_1 + \lambda_2 + \mu_1 + \mu_2 - \lambda_2\Delta_0^B) + \frac{2\lambda_1\mu_2}{2\mu_2 - \lambda_2\Delta_0^B}) \end{bmatrix}.$$

The MC in Figure 4.6 tracks the number of Class-1 jobs present when a Level-0 Class-2 busy period starts; $\Pi_i, \forall i \geq 0$ is the solution to this MC. To find the Π_i, we write down the

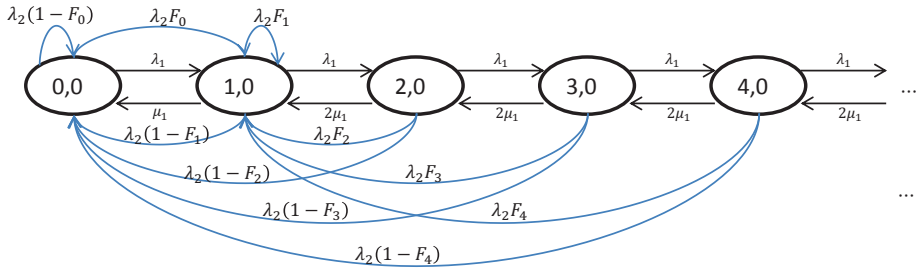

Figure 4.6: The MC when there are no Class-2 jobs.

Balance Equations:

$$\lambda_2(1-\Pi_0) = \lambda_1\Pi_0 - \mu_1\Pi_1 + \lambda_2\varphi_1 \tag{4.51}$$

$$\lambda_2(1-\Pi_0-\Pi_1) = \lambda_1\Pi_1 - 2\mu_1\Pi_2 \tag{4.52}$$

$$\vdots$$

$$\lambda_2(1-\sum_{j=0}^{i}\Pi_j) = \lambda_1\Pi_i - 2\mu_1\Pi_{i+1} \tag{4.53}$$

Again, using the same technique, we can express Π_i.

Lemma 5 *Π_i can be expressed as a function of φ_1:*

$$\Pi_i = \begin{cases} \frac{\mu_1(1-f)+\lambda_2(1-\varphi_1)}{\lambda_1+\lambda_2+\mu_1-f\mu_1} & \text{for } i=0 \\ \frac{(1-f)(\lambda_1+\lambda_2\varphi_1)}{\lambda_1+\lambda_2+\mu_1-f\mu_1}f^{i-1} & \text{for } i\geq 1 \end{cases}, \tag{4.54}$$

where $f = \frac{1}{4\mu_1}(\lambda_1+\lambda_2+2\mu_1-\sqrt{(\lambda_1+\lambda_2+2\mu_1)^2-8\lambda_1\mu_1})$.

Substituting (4.50) and (4.54) in (4.43) gives us an equation of φ_1, from which we can get φ_1:

$$\varphi_1 = \frac{\lambda_1(f-1)E + \Delta_0^F\frac{2\lambda_1}{2\mu_2-\lambda_2\Delta_0^B}(\lambda_2+\mu_1-f\mu_1)}{-\lambda_2(f-1)E + \Delta_0^F\frac{2\lambda_1\lambda_2}{2\mu_2-\lambda_2\Delta_0^B} + (\lambda_1+\lambda_2+\mu_1-f\mu_1)}, \tag{4.55}$$

where $E = -\frac{1}{f-1}(\frac{g\xi_2}{g-1}+\frac{h\xi_1}{h-1}-\frac{\Delta_0^F(2\lambda_1+2\mu_2-\lambda_2\Delta_0^B)}{2\mu_2-\lambda_2\Delta_0^B}) + \frac{g\xi_2}{(fg-1)(g-1)} + \frac{h\xi_1}{(fh-1)(h-1)}$.

Hence, (4.41), (4.42) and (4.55) give four equations with four unknowns whose solution gives $\vec{d_0}$.

4.5.4 Summary and Guidelines for $c \geq 2$ case

The process to get $G_{L^2}(z)$ can be easily extended to $c > 2$ case. We summarize the main steps here:

- Transform the 2D-infinite continuous-time MC (in Figure 4.1) into a 1D-infinite ladder-like discrete-time EMC (with the transition matrix in (4.23)) by:
  1. using the Class-1 busy period to simplify the original MC to the MC in Figure 4.2.
  2. deriving the transition matrix of the EMC by observing the system state at Class-2 departures; deriving $M_{i\to j}$ $(1 \leq i \leq c+1)$ for three cases: $L_k^2 \geq c$, $L_k^2 \in [1, c-1]$ and $L_k^2 = 0$ as done in Section 4.5.1; and inserting $M_{i\to j}$ into M according to (4.22).
- Derive the closed-form expression of the GF of the number of Class-2 jobs in the system. The matrix in (4.23) has a nice structure and we can apply the GF approach to express $(G_{(0,L^2)}(z), \ldots, G_{(c-1,L^2)}(z))$ as the product of a row vector $\vec{d_0} = (d_{00}, ..., d_{c-1,0})$ and a $(c \times c)$ matrix $\mathscr{D}(z)$. Then, we use linear recurrence sequence (matrix difference equations if $c > 2$) to get equations like (4.43). The solution of these equations gives φ_i for $i = 1, ..., c-1$. Once we obtain $G_{(i,L^2)}(z)$ in closed form for $i \in [0, c-1]$, we use (4.24) to derive the GF of L^2.

4.6 Numerical Method

While it is theoretically possible to get the $G_{L^2}(z)$ in closed form for any $c > 2$, the expressions of $G_{L^2}(z)$ are not simple even for $c = 2$, and become more complicated when c increases, because:

1. Expressing the transition matrix for $L_k^2 \geq c$ requires deriving the LTs for c^2 different D_k, depending on c^2 different combinations of L_k^1 and L_{k+1}^1.
2. Expressing $\mathscr{D}(z)$ as in (4.61) requires an elaborate derivation.
3. Deriving a closed-form expression for φ_i for $i = 1, ..., c-1$ requires solving matrix difference equations.

We next propose an easily implementable numerical algorithm to calculate $E\left[L^2\right]$ which overcomes these difficulties. For generality, we focus on $E\left[L^2\right]$ and $E\left[R_2\right]$ instead of the distribution of R_2, which requires the FIFO assumption.

To overcome difficulty 1, we give a numerical algorithm to calculate $\mathscr{A}_i$, the probability of $i = 0, 1, ...$ Class-2 arrivals during different inter-departure times, in Section 4.5.1. Then, the techniques in Subsections 4.5.1 and 4.5.1 can be used to derive the transition matrix of the EMC for $L_k^2 \geq c$.

We demonstrate the algorithm for expressing $\mathscr{A}_i$ in Appendix 4.10.2 by deriving the transition probabilities of the EMC for $L_k^2 \geq c = 2$. The general case with $c > 2$ can be analyzed similarly.

Once we get $\mathscr{A}_i$, using (4.30), we obtain the rows of M in (4.23) that correspond to any S_i with $i \geq 2$ numerically. Then, using (4.35) and (4.36), we can compute the transition matrix in (4.23).

Since we cannot practically store an infinite number of matrices, we store up to $Limit \times (c+1)$ matrices of dimension $c \times c$, given that the maximum element of $\mathscr{A}_{Limit}$ is less than the accuracy tolerance, i.e., $\max(\mathscr{A}_{Limit}) < Tolerance$. Therefore, these matrices accurately capture the behavior of the whole system when the $Tolerance$ is small enough.

Likewise, for any $c > 2$, we can derive the transition matrix of the EMC in (4.23) numerically, by discussing the three cases: $L_k^2 \geq c$, $L_k^2 \in [1, c-1]$, and $L_k^2 = 0$. In this way, we efficiently derive the transition matrix for $L_k^2 \geq c$, which requires expressing the LTs for c^2 different D_k.

To overcome difficulties 2 and 3, we use numerical methods to solve the 1D-infinite EMC in (4.23). Riska and Smirni (2002) gives an exact aggregate method to derive different moments of the number of Class-2 jobs in the system. This method is easy to implement using their Theorem 3.1, (18) and (21). As an example, we derive the first moment. See Algorithm 4 in Appendix 4.10.4. This numerical procedure is the basis for our results in Section 4.7.

4.7 Numerical Results and Extensions

This section reports on a set of numerical results using Algorithm 4 (denoted by ^{N}). We will validate our results in two cases where exact results are available (i.e., when $c = 2$ and when

$\mu_1 = \mu_2$) and show that the relative errors are small. (We denote the exact results generated in these two cases by ^{EX}.) Next, we will apply our numerical results to answer questions of interest for multi-server queue with prioritization. Finally, we apply our methodology to the problem in Maglaras and Zeevi (2004) by replacing Class-1 BP in our model with Class-1 jobs' exponential service time.

Throughout this section, we use $\lambda_i = c\rho_i\mu_i$ for $i = 1, 2$, so that $\rho_1 + \rho_2 < 1$ is each server's occupation rate in the $M/M/c$ queue. Thus once c, ρ_1, ρ_2, μ_1 and μ_2 are given, the system is determined.

4.7.1 Accuracy and Complexity of the Proposed Numerical Method

Potential inaccuracies in Algorithm 4 arise from two main sources. The first is that α_l^{BP} requires numerical inversion of the probability GF. Abate and Whitt (1992) gave an efficient inversion algorithm with a controllable error bound. We use the suggested error bound: 10^{-8} in Algorithm 4. The second inaccuracy is controlled by the tolerance we choose when assuming a finite waiting room for Class-2 jobs: $Limit = \min\{i \mid \max(\mathscr{A}_i) \leq Tolerance\}$ where $\mathscr{A}_i$s are given in (4.66). We use $Tolerance = 10^{-6}$.

We demonstrate the accuracy of Algorithm 4 by comparing the expected L^2, generated using Algorithm 4, $E^N[L^2]$, with the exact results for $c = 2$ and for $\mu_1 = \mu_2 = 1$. We denote the relative error of $E^N[L^2]$ in comparison with $E^{EX}\left[L^2\right]$ as $\%Error = \frac{\left|E^{EX}\left[L^2\right] - E^N\left[L^2\right]\right|}{E^{EX}[L^2]}$.

In the $c = 2$ case, we calculate $E^{EX}[L^2]$ using the closed form $G_{L^2}(z)$ from Section 4.5, for $\mu_1 = 1$ and $\mu_2 = 2$. In the $\mu_1 = \mu_2 = 1$ case, we calculate $E^{EX}[L^2]$ using Buzen and Bondi (1983):

$$E^{EX}\left[L^2\right] = \frac{(\lambda_1+\lambda_2)^{c+1}}{c!(\frac{c-\lambda_1-\lambda_2}{c}\sum_{n=0}^{c-1}\frac{(\lambda_1+\lambda_2)^n}{n!} + \frac{(\lambda_1+\lambda_2)^c}{c!})(c-\lambda_1-\lambda_2)} - \frac{\lambda_1^{c+1}}{c!(\frac{c-\lambda_1}{c}\sum_{n=0}^{c-1}\frac{\lambda_1^n}{n!} + \frac{\lambda_1^c}{c!})(c-\lambda_1)} + \lambda_2,$$

for $c = 2, 10, 50$. In both cases, we vary $\rho = \rho_1 + \rho_2$ (from 0.90 to 0.99 in steps of 0.01) and $\frac{\rho_1}{\rho_2}$ (as $\frac{1}{9}, \frac{1}{4}, \frac{2}{3}, 1, \frac{3}{2}, \frac{4}{1}, \frac{9}{1}$). In total, we exam 280 different parameters settings, and all $\%Error$s in these experiments are less than 0.001%, outperforming the approximation in Harchol-Balter et al. (2005), which is to our knowledge the best approximation, with a $\%Error$ within 2% compared to simulation.

Next, we discuss the complexity of Algorithm 4 regarding the combination of parameters, i.e., c, ρ_1, ρ_2, μ_1, and μ_2 (note that, λ_1 and λ_2 are determined by these parameters: $\lambda_i = c\rho_i\mu_i$ for $i = 1, 2$). Specially, we discuss how the time for Algorithm 4 to generate $E^N[L^2]$ changes when we *solely* increase one of the parameters.

The complexity of the algorithm increases with c. Recall that the size of each block matrix in (4.37) is $c \times c$. When c increases, the algorithm needs more storage space and computing power to handle a larger matrix. For $c \leq 50$, it only takes several seconds. The processing time of Algorithm 4 increase with c. For $c = 100$, it may take several minutes. Detailed numerical results are available upon request.

Increasing ρ_1 raises the complexity of the algorithm. As we know, the expected Class-1 busy period is $\frac{1}{c\mu_1(1-\rho_1)}$ (see, e.g., Adan and Resing 2002). Thus, a larger ρ_1 causes a stochastically longer Class-1 busy period, within which more Class-2 customers may arrive. If we imagine different arrows in Figure 4.2 as liquid flows, we could expect a higher probability of seeing the Markov Chain at a state with larger q_2. Then, our algorithm needs more iterations to reach a given tolerance. The above discussion also demonstrates that $E[L^2]$ increases with ρ_1 too. Similar discussion will lead to that increasing μ_1 reduces the complexity of the algorithm and $E[L^2]$ at the same time.

It is straightforward that the complexity of the algorithm and $E[L^2]$ both increase with ρ_2. Again a busier system implies that more iterations are required to reach a given tolerance. However, it is not obvious that the complexity of the algorithm increases with μ_2. Since $\rho_2 = \frac{\lambda_2}{c\mu_2}$ is fixed, we need to raise λ_2 when increasing μ_2 to keep ρ_2 the same. A higher λ_2 leads to more Class-2 arrivals in the Class-1 busy period. The rest of the discussion is exactly the same as the case of increasing ρ_1.

Given the accuracy of Algorithm 4, we next use it to derive insights on the operation of multi-server queueing systems with two priority classes. None of these insights were available before due to the lack of exact numerical algorithms for this preemptive system with different service rates for each priority class.

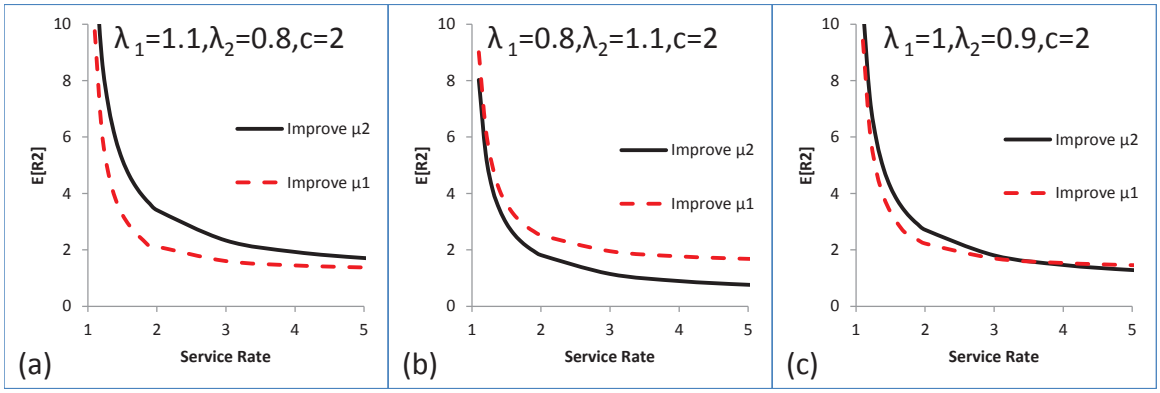

Figure 4.7: The effect of improving $\mu_i, i = 1, 2$ on $E[R_2]$ under different combinations of λ_i for $i = 1, 2$.

4.7.2 Insight 1 - How Changing μ_1 or μ_2 Affects $E[R_2]$

Consider a company that operates an $M/M/2$ system to serve two priority classes where Class-1 has preemptive priority over Class-2. The company receives complaints of long response times from Class-2 customers. When the manager is able to improve the service rate of either one priority class, her first reaction may be to improve the service rate of Class-2 customers, μ_2. However, this decision may not be optimal: Consider the example with $\lambda_1 = 1.1$, $\lambda_2 = 0.8$ and $\mu_1 = \mu_2 = 1$. Figure 4.7(a) illustrates the effect of improving μ_1 or μ_2 on $E[R_2]$. The solid line shows how $E[R_2]$ changes when improving μ_2 while keeping $\mu_1 = 1$ and the dashed line shows the effect on $E[R_2]$ of improving μ_1 while fixing $\mu_2 = 1$: for the same service rate improvement, upgrading μ_1 is more effective in reducing $E[R_2]$. In other words, when Class-2 customers complain about the long response time they experience, it is better to improve the service rate of Class-1 customers. The intuition is as follows.

Any Class-2 customer's response time is dictated by its interaction with customers of both types. Class-1 customers affect this time by the service time of Class-1 customers seen upon a Class-2 customer's arrival, those Class-1 customers who arrive during her waiting time if she does not enter service immediately, and those Class-1 jobs who interrupt her service. Class-2 customers affect this time via both the service times of customers at her arrival and her own service time. Increasing μ_1 reduces the first part, while increasing μ_2 reduces the second part. Which of these effects dominates (and which service rate is preferable to improve) depends on the relation between λ_1 and λ_2. In Figure 4.7(a), the first part of response time dominates,

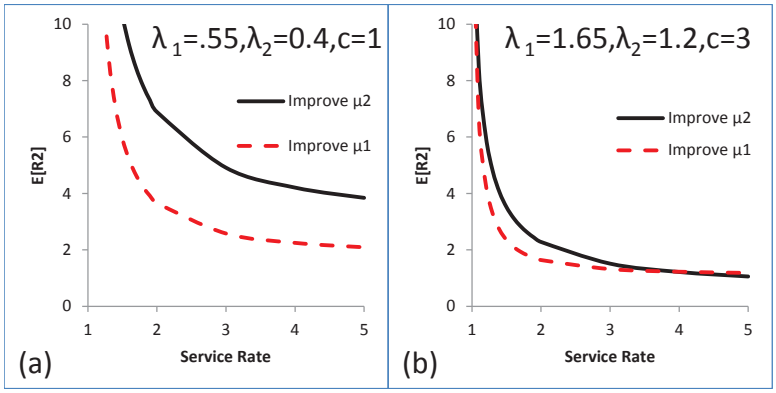

Figure 4.8: The effect of improving $\mu_i, i = 1, 2$ on $E[R_2]$ under different numbers of servers.

so it is better to improve μ_1. In contrast, when $\lambda_1 = 0.8$ and $\lambda_2 = 1.1$, as in Figure 4.7)(b), improving μ_2 is better. Finally, when $\lambda_1 = 1$ and $\lambda_2 = 0.9$, from Figure 4.7(c), we see that it is better to improve μ_1, if the maximum service rate the company can achieve is below 3.5; otherwise it is better to improve μ_2. Therefore, we need an accurate calculation of $E[R_2]$ to decide which service rate to improve.

Next, we examine how the number of servers may affect the manager's decision. In Figures 4.8 (a) and (b), we keep the initial service and congestion rates for both classes the same (i.e. $\mu_1 = \mu_2 = 1$, $\rho_1 = 0.55$, and $\rho_2 = 0.4$), and change c (i.e., $c = 1$ in 4.8(a), $c = 2$ in 4.7(a), and $c = 3$ in 4.8(b). We see that when c increases, the two $E[R_2]$ curves get closer, and when $c = 3$, they cross. This example illustrates that managers cannot decide on which service rate to improve by approximating an $M/M/c$ system as an $M/M/1$, because the number of servers affects this decision.

Other objective functions besides $E[R_2]$ can also be considered by our algorithm. For example, we can consider a weighted average of $E[R_1]$ and $E[R_2]$. Because we can calculate $E[R_2]$ quickly and accurately, comparing different service rates of different priority classes is straightforward. Likewise, it is very simple to incorporate different marginal costs of improving different service rates. Of course, we could also consider the case where the company can improve μ_1 and μ_2 simultaneously.

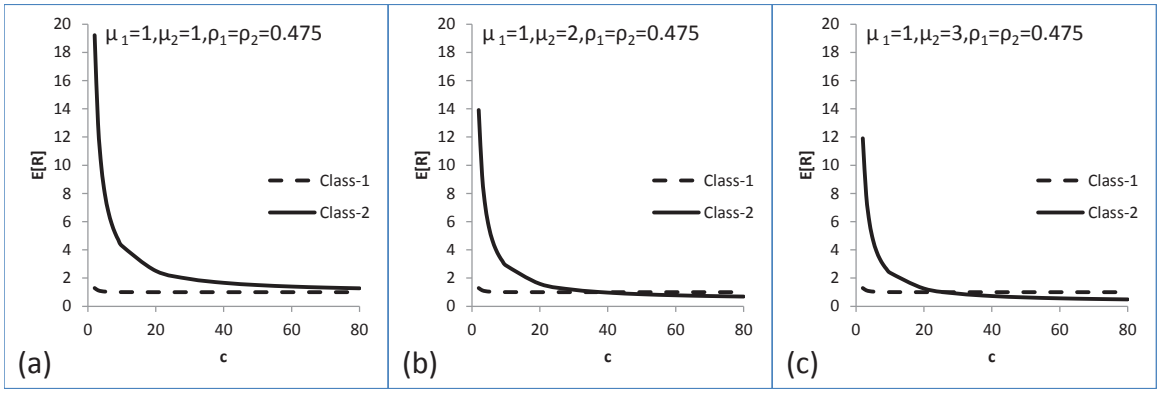

Figure 4.9: The Marginal Effect of pooling servers on expected waiting times of both priority classes, under different μ_i for $i = 1, 2$ when $\rho_1 = \rho_2 = 0.475$.

4.7.3 Insight 2 - The Marginal Effect of Pooling Servers

In this section, we consider the effect of increasing c, while keeping the occupation rates ρ_i and service rates of both priority classes μ_i $(i = 1, 2)$ the same, i.e., the marginal effect of pooling servers. In Figures 4.9 (a-c), we illustrate the effect of pooling c servers together on the expected response times of Class-2 jobs, under different μ_2's. We use $\rho_1 = \rho_2 = 0.475$ in this section.

The first observation is that pooling servers always reduces expected response times of both priority classes; this is no surprise. As more servers are added, the response times approach $\frac{1}{\mu_i}$, respectively, for Class-1 and Class-2 jobs; in the infinite servers case, the response times are simply $\frac{1}{\mu_i}$. Note though that adding servers always benefits Class-2 jobs more than it benefits Class-1 jobs. The intuition is that due to their preemptive priority, Class-1 jobs already have the highest access to servers, while Class-2 jobs' access is restricted. Also, comparing Figures 4.9 (a-c) reveals that when μ_2 increases, the relative improvement in $E[R_2]$ is reduced. The reason is that shorter Class-2 jobs have a greater chance to finish before being interrupted by a Class-1 arrival. Therefore, pooling servers aids long Class-2 jobs more than short Class-2 jobs. The same insight holds for different ρ's and $\frac{\rho_1}{\rho_2}$'s.

4.7.4 Insight 3 - Few Fast Servers v.s. Many Slow Servers

In this section we compare systems with different numbers of servers, while keeping the arrival rates λ_i and the occupation rates ρ_i $(i = 1, 2)$ the same (i.e., increase c and reduce μ_i while holding $c\mu_i = \frac{\lambda_i}{\rho_i}$ constant). That is we investigate the effect of having many slow servers

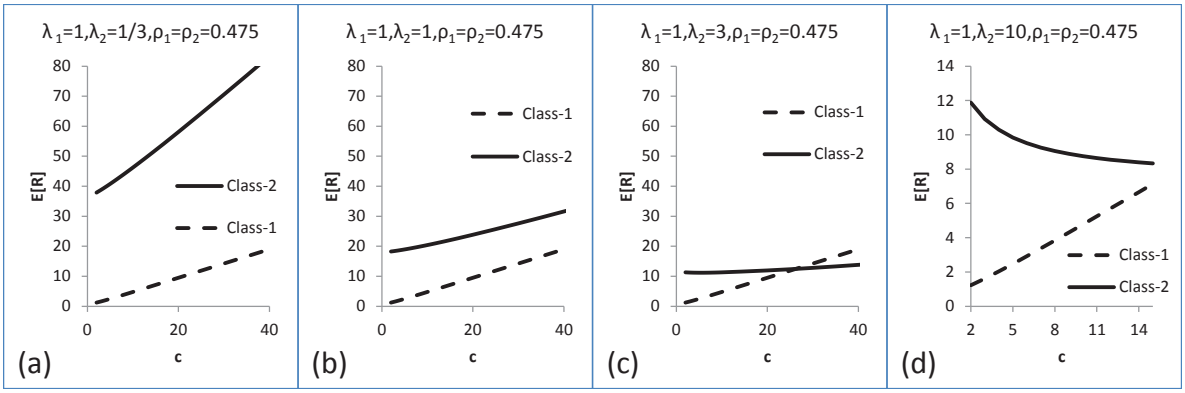

Figure 4.10: The effect of c on expected waiting times of both priority classes, under different $\lambda_i, i = 1, 2$.

compared with having fewer fast servers. We again use $\rho_1 = \rho_2 = 0.475$.

In Figure 4.10, we fix $\lambda_1 = 1$ and illustrate the effect of having more slow servers on the expected response times of Class-2 jobs, under different λ_2's. Note that within each figure as c increases, both μ_1 and μ_2 decrease. Between figures, because we keep $\frac{\lambda_2}{c\mu_2} = \rho_2 = 0.475$, for the same c, a smaller λ_2 results in smaller μ_2 and vice versa.

We see that in (almost) all cases jobs prefer fewer fast servers, which might be expected, as the service times are exponential ($cv = 1$). But we also see that the number of servers affects $E[R_2]$ in different manners for different values of λ_2. When $\lambda_2 = \frac{1}{3}$, Class-2 response times increase faster than Class-1 jobs' as c increases. In contrast, when $\lambda_2 = 3$, Class-2 response times increase slower than Class-1 jobs' as c increases. The intuition is that even though reducing μ_2 increases Class-2 response times due to Class-2 service time, higher c provides Class-2 jobs more access to servers, so they have a higher chance to finish before being interrupted by a Class-1 arrival. When $\lambda_2 = 1$, these two factors balance and the response times of both priority classes increase with c at similar rates. When Class-2 jobs are short, the increased access is more beneficial as they are more likely to finish before being interrupted.

Another observation from Figures 4.10 (a-c) is that when $\lambda_1 = 1$ and $\lambda_2 = 3$, the Class-2 jobs' average response time may decrease with c, when c is small. This trend is more obvious in Figure 4.10 (d) when $\lambda_1 = 1$ and $\lambda_2 = 10$: the $E[R_2]$ decreases by about a third (12 vs 8) when c increases from 2 to 14. In this case, the effect of improved access to servers benefits Class-2 jobs so much that it overcompensates for the negative effect of decreasing μ_2. This result has

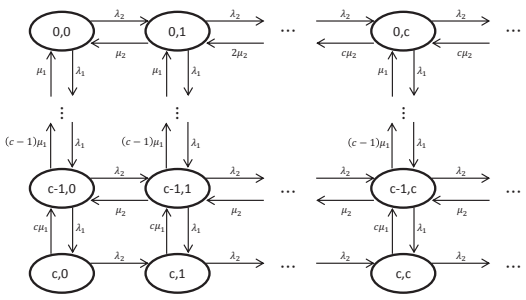

Figure 4.11: The MC of an $M/M/c$ queue with two priority classes where the first class is completely impatient.

also been shown in Wierman et al. (2006). Our numerical algorithm strengthens their result and gives an accurate estimation of the optimal number of servers for low-priority customers. The same insights hold for different ρ's and $\frac{\rho_1}{\rho_2}$'s.

4.7.5 Extension to Impatient Class-1 Jobs

Maglaras and Zeevi (2004) considered an $M/M/c$ queue with two priority classes where the first class is completely impatient, i.e., if not served at arrival, they leave the system. They applied diffusion approximations to the problem in the asymptotic Halfin and Whitt (1981) regime. The MC of their problem is given in Figure 4.11, and is similar to the MC in Figure 4.1, except that it is truncated at $q_1 = c$ and is a 1D-infinite MC.

Our methodology can be applied to this system by directly replacing the Class-1 BP in our model with the $\exp(c\mu_1)$ distributed busy periods caused by Class-1 jobs. Therefore, we can obtain a closed-form expression of the GF of L^2 when $c = 2$; and we have an efficient numerical algorithm to calculate the distribution of L^2 when $c \geq 2$.

Table 4.1 illustrates the accuracy of the two approximations (2D diffusion and perturbation) in Maglaras and Zeevi (2004) and of Algorithm 4 in our paper, compared with simulation under different settings in their paper. For simulation, 2D diffusion and perturbation approximations, we generate the $E[L^1 + L^2]$ from the distribution of $L^1 + L^2$ (i.e., the total number of jobs in the system) in their results. (Unfortunately, the confidence intervals of the simulation was not provided.) We generate $E[L^1 + L^2]$ from the sum of $E[L^1]$, obtained using a single-class $M/M/c/c$ model (page 81, Gross et al. 2008), and $E[L^2]$, obtained using Algorithm 4. The results of our algorithm are well within the margin of errors of their simulation and are typically

closer to the simulation than their two approximations. The noticeable errors of our algorithm may be within the inaccuracy of the simulation. Note that Maglaras and Zeevi's approximations are only accurate in the Halfin and Whitt regime, i.e., for high ρ and c, whereas our method is accurate for all such combinations. However, the time consumption of our algorithm increases with c. When $c = 150$, the processing time of our algorithm raises up to 30 minutes.

	Simulation	2D Diffusion		Perturbation		Our Algorithm	
(c, ρ, μ_1, μ_2)	$E\left[L^1 + L^2\right]$	$E[L^1 + L^2]$	$\%Error$	$E[L^1 + L^2]$	$\%Error$	$E[L^1 + L^2]$	$\%Error$
(100,0.95,1,2)	108.22	109.28	1.0%	112.34	3.8%	108.42	0.2%
(50,0.95,1,2)	65.71	64.88	1.3%	66.82	1.7%	64.57	1.7%
(150,0.95,1,2)	154.49	154.30	0.1%	158.63	2.7%	153.58	0.6%
(100,0.925,1,2)	99.64	98.02	1.6%	102.50	2.9%	98.13	1.5%
(100,0.975,1,2)	140.36	136.57	2.7%	139.77	0.4%	138.42	1.4%
(100,0.95,1,5)	120.46	120.02	0.4%	119.43	0.9%	118.97	1.2%
(100,0.95,2,1)	102.74	103.24	0.5%	111.39	8.4%	102.60	0.1%
(100,0.95,5,1)	101.49	101.16	0.3%	115.83	14.1%	101.31	0.2%
(100,0.95,20,10)	103.44	103.39	0.1%	112.27	8.5%	102.60	0.8%

Table 4.1: 2D Diffusion and Perturbation in Maglaras & Zeevi 2004 v.s. Algorithm 4 in terms of $E[L^1 + L^2]$ for different settings with $\rho_1 = \rho_2 = \frac{\rho}{2}$.

4.8 Summary

This paper analyzed an $M/M/c$ queue with two preemptive-resume priority classes. This problem is usually described by a 2-dimension infinite MC, representing the two class state space. We introduced a new technique to reduce this 2D-infinite MC into a 1D-infinite MC, from which the Generating Function (GF) of the number of low-priority jobs can be derived in closed form. We demonstrate this methodology for the $c = 1, 2$ cases. When $c > 2$, the closed-form expression of the GF becomes cumbersome. We thus derive an exact numerical algorithm to calculate different moments of the number of Class-2 jobs in the system for any $c \geq 2$.

Interesting insights are derived using our algorithm: We first showed that for a company serving two priority classes and receiving complaints of long response times from Class-2 customers, the manager may wish to improve the service rate of Class-1 customers under certain conditions. Secondly, we showed that pooling servers always benefits Class-2 jobs more than Class-1 jobs, and aids long Class-2 jobs more than short Class-2 jobs. Thirdly, we demonstrated

that even though in the single-class system a few fast servers are always preferred to many slow servers, in a system with priority Class-2 jobs may prefer many slow servers to a few fast servers. Finally, we applied our methodology to the problem considered by Maglaras and Zeevi (2004), and demonstrated that our algorithm is more accurate than their approximations.

For future research, it would be very beneficial to extend our methodology to more than two priority classes, though this appears to be quite challenging. As priority queues have a direct application in information and communication services, it would be interesting to incorporate pricing and system design into the model and try to maximize profit.

4.9 References

Abate, J., W. Whitt. 1992. Numerical Inversion of Probability Generating Functions. *Operations Research Letters* **12**(4) 245-251.

Abouee-Mehrizi, H., B., Balcıoglu, O. Baron. 2012. Strategies for a Centralized Single Product Multi-Class M/G/1 Make-to-Stock Queue. *Operations Research* **60**(4) 803-812.

Bertsimas, D., D. Nakazato. 1995. The distributional Little's Law and its applications. *Operations Research* **43**(2) 298-310.

Buzacott, J., J. Shanthikumar. 1993. Stochastic Models of Manufacturing Systems. *Prentice Hall.*

Buzen J., A. Bondi. 1983. The response times of priority classes under preemptive resume in $M/M/m$ queues. *Operations Research* **31**, 456-465.

Davis, R. 1966. Waiting-time distribution of a multi-server, priority queueing system. *Operations Research 14*(1) 133-136.

Green, D., D. Knuth. 1990. *Mathematics for the Analysis of Algorithms*, 3rd ed. Birkhāuser Boston.

Gross, D., J. Shortle, J. Thompson, C. Harris. 2008. *Fundamentals of Queueing Theory.* Wiley & Sons.

Halfin, S., W. Whitt. 1981. Heavy-traffic Limits for Queues with Many Exponential Servers. *Operations Research* **29**(3) 567-588.

Harchol-Balter, M., T. Osogami, A. Scheller-Wolf, A. Wierman. 2005. Multi-server queueing systems with multiple priority classes. *Queueing Systems* **51**(3) 331–360.

Jeffrey, A. 2005. *Complex Analysis and Applications.* 2nd ed. CRC.

Kella O., U. Yechiali. 1985. Waiting time in the non-preemptive priority $M/M/c$ queue. *Stochastic Models* **1**(2) 257-262.

Maglaras, C., A. Zeevi. 2004. Diffusion Approximations for a Multiclass Markovian Service System with "Guaranteed" and "Best-Effort" Service Levels. *Math. of Operations Research* **29**(4) 786-813.

Maglaras, C., A. Zeevi. 2005. Pricing and Design of Differentiated Services: Approximate Analysis and Structural Insights. *Operations Research* **53**(2) 242-262.

Riska, A., E. Smirni. 2002. Exact Aggregate Solutions for M/G/1-type Markov Processes, *SIGMETRICS* 86-96.

Takagi, H. 1991. *Queueing analysis: a foundation of performance evaluation.* Amsterdam, North-Holland.

Van Mieghem, J. 1995. Dynamic Scheduling with Convex Delay Costs: The Generalized $c\mu$ Rule. *The Annals of Applied Probability* **5**(3) 808-833.

Wierman, A., M. Harchol-Balter, T. Osogami, A. Scheller-Wolf. 2006. How Many Servers are Best in a Dual-Priority M/PH/k system. *Performance Evaluation* **63**(12) 1253-1272.

4.10 Appendix

4.10.1 Calculations

Calculation for $G_{\hat{L}^2}(z)$

The following result will be used in the calculation for $G_{\hat{L}^2}(z)$. The derivation of them is straightforward, so we skip all the details.

$$\sum_{n=0}^{\infty} \begin{bmatrix} \alpha_n^{\hat{D}_k} & \cdots & \alpha_1^{\hat{D}_k} & \alpha_0^{\hat{D}_k} & \mathbf{0}_{1\times\infty} \end{bmatrix}^T z^n = \begin{bmatrix} 1 & z & z^2 & z^3 & \cdots \end{bmatrix}^T G_{\alpha^{\hat{D}_k}}(z).$$

With the help of these results, we derive $G_{\hat{L}^2}(z)$:

$$\begin{aligned}
G_{\hat{L}^2}(z) &= (\begin{bmatrix} \hat{d}_1, \hat{d}_2, ... \end{bmatrix} + \hat{d}_0 \hat{\Psi}_{01}) \sum_{n=0}^{\infty} \begin{bmatrix} \alpha_n^{\hat{D}_k} & \cdots & \alpha_1^{\hat{D}_k} & \alpha_0^{\hat{D}_k} & \mathbf{0}_{1\times\infty} \end{bmatrix}^T z^n, \\
G_{\hat{L}^2}(z) &= \begin{bmatrix} \hat{d}_1, \hat{d}_2, ... \end{bmatrix} \begin{bmatrix} 1 & z & z^2 & z^3 & \cdots \end{bmatrix}^T G_{\alpha^{\hat{D}_k}}(z) + \hat{d}_0 \hat{\Psi}_{01} \begin{bmatrix} 1 & z & z^2 & z^3 & \cdots \end{bmatrix}^T G_{\alpha^{\hat{D}_k}}(z), \\
G_{\hat{L}^2}(z) &= \frac{G_{\hat{L}^2}(z) - \hat{d}_0}{z} G_{\alpha^{\hat{D}_k}}(z) + \frac{\hat{d}_0}{z} \frac{z\lambda_2 + \lambda_1 G_{\alpha^{BP}}(z) - \alpha_0^{BP}\lambda_1}{\lambda_1 + \lambda_2 - \alpha_0^{BP}\lambda_1} G_{\alpha^{\hat{D}_k}}(z).
\end{aligned}$$

We move $G_{\hat{L}^2}(z)$ to the left-hand side and get (4.19).

Calculation for $G_{L^2}(z)$

The following results will be used in the calculation for $\mathscr{D}(z)$. The derivation of them is straightforward, so we skip all the details.

$$\sum_{i=1}^{\infty} \begin{bmatrix} \mathscr{A}_{n-1}^T & \cdots & \mathscr{A}_1^T & \mathscr{A}_0^T & \mathbf{0}_{1\times\infty} \end{bmatrix}^T z^i = z\Upsilon G_{\mathscr{A}}, \tag{4.56}$$

$$\text{in which } \Upsilon = \begin{bmatrix} I_{2\times2} & zI_{2\times2} & z^2I_{2\times2} & z^3I_{2\times2} & \cdots \end{bmatrix}^T \text{ and } G_{\mathscr{A}} = \begin{bmatrix} G_{\alpha^{00}}(z) & G_{\alpha^{01}}(z) \\ G_{\alpha^{10}}(z) & G_{\alpha^{11}}(z) \end{bmatrix}.$$

$$\left[\vec{d_2}, \vec{d_3}, ...\right] z\Upsilon = \frac{1}{z}(G_{L^2}(z) - \vec{d_0} - \vec{d_1}z) \tag{4.57}$$

$$d_1 = d_0(\Psi_{10}^{-1} - \Psi_{01}) \tag{4.58}$$

$$\begin{aligned} z^2\Psi_{10}^{-1}\Psi_{12}\Upsilon &= \frac{z^2\Psi_{10}^{-1}\begin{bmatrix} \lambda_2(\lambda_1+\lambda_2+\mu_1+\mu_2-\alpha_0^B\lambda_1) & \frac{1}{z}\lambda_1(z\lambda_2+\lambda_1G_{\alpha^B}(z)-\alpha_0^B\lambda_1) \\ \lambda_2\mu_1 & \frac{1}{z}(\lambda_1+\lambda_2+\mu_2)(z\lambda_2+\lambda_1G_{\alpha^B}(z)-\alpha_0^B\lambda_1) \end{bmatrix}}{(\lambda_1+\lambda_2+\mu_2)(\lambda_1+\lambda_2+\mu_2-\lambda_1\alpha_0^B)+\mu_1(\lambda_2+\mu_2)} \\ &= \begin{bmatrix} z^2\frac{\lambda_2}{\mu_2} & 0 \\ 0 & \frac{z}{\mu_2}(z\lambda_2+\lambda_1G_{\alpha^B}(z)-\alpha_0^B\lambda_1) \end{bmatrix} \end{aligned} \tag{4.59}$$

$$\Psi_{02}\Upsilon = \begin{bmatrix} 0 & \frac{1}{z^2}\frac{\lambda_1^2(G_{\alpha^B}(z)-\alpha_0^B-z\alpha_1^B)}{\lambda_1^2+\lambda_2^2-\alpha_0^B\lambda_1^2+2\lambda_1\lambda_2+\lambda_2\mu_1-\alpha_0^B\lambda_1\lambda_2} \\ 0 & \frac{1}{z^2}\frac{\lambda_1(\lambda_1+\lambda_2)(G_{\alpha^B}(z)-\alpha_0^B-z\alpha_1^B)}{\lambda_1^2+\lambda_2^2-\alpha_0^B\lambda_1^2+2\lambda_1\lambda_2+\lambda_2\mu_1-\alpha_0^B\lambda_1\lambda_2} \end{bmatrix} \tag{4.60}$$

With the help of these results, we derive $\mathscr{D}(z)$:

$$\left[G_{(0,L^2)}(z), G_{(1,L^2)}(z)\right] = \vec{d_0}+(\left[\vec{d_2}, \vec{d_3}, ...\right]+\vec{d_1}\Psi_{12}+\vec{d_0}(\Psi_{01}\Psi_{12}+\Psi_{02}))\sum_{n=1}^{\infty}\left[\mathscr{A}_{n-1}^T \ \cdots \ \mathscr{A}_1^T \ \mathscr{A}_0^T \ \mathbf{0}_{1\times\infty}\right]^T z^n.$$

From (4.56), we have

$$\left[G_{(0,L^2)}(z), G_{(1,L^2)}(z)\right] = \vec{d_0} + z\left\{\left[\vec{d_2}, \vec{d_3}, ...\right] + \vec{d_1}\Psi_{12} + \vec{d_0}(\Psi_{01}\Psi_{12} + \Psi_{02})\right\}\Upsilon G_{\mathscr{A}}.$$

From (4.57), we have

$$[G_{(0,L^2)}(z), G_{(1,L^2)}(z)] = \vec{d_0}+\frac{1}{z}([G_{(0,L^2)}(z), G_{(1,L^2)}(z)]-\vec{d_0}-\vec{d_1}z)G_{\mathscr{A}}+z(\vec{d_1}\Psi_{12}+\vec{d_0}(\Psi_{01}\Psi_{12}+\Psi_{02}))\Upsilon G_{\mathscr{A}}.$$

Moving $\left[G_{(0,L^2)}(z), G_{(1,L^2)}(z)\right]$ to the left side of the equation gives

$$\left[G_{(0,L^2)}(z), G_{(1,L^2)}(z)\right](zI_{2\times2}-G_{\mathscr{A}}) = \vec{d_0}(z^2(\Psi_{01}\Psi_{12}+\Psi_{02^+})\Upsilon G_{\mathscr{A}}-G_{\mathscr{A}}+zI_{2\times2})+\vec{d_1}(z^2\Psi_{12}\Upsilon G_{\mathscr{A}}-zG_{\mathscr{A}}).$$

From (4.58), we have

$$\left[G_{(0,L^2)}(z), G_{(1,L^2)}(z)\right](zI_{2\times 2}-G_{\mathscr{A}}) = \vec{d_0}\left\{(z^2\Psi_{10}^{-1}\Psi_{12}\Upsilon + z^2\Psi_{02}\Upsilon - I_{2\times 2} - z(\Psi_{10}^{-1}-\Psi_{01}))G_{\mathscr{A}} + zI_{2\times 2}\right\}.$$

From (4.59) and (4.60), we have

$$\begin{aligned}
&\left[G_{(0,L^2)}(z), G_{(1,L^2)}(z)\right](zI_{2\times 2} - G_{\mathscr{A}}) \\
= \quad & \vec{d_0}\left(\begin{bmatrix} z^2\frac{\lambda_2}{\mu_2} - 1 & \lambda_1^2\frac{(G_{\alpha^B}(z)-\alpha_0^B - z\alpha_1^B)}{\lambda_1^2+\lambda_2^2-\alpha_0^B\lambda_1^2+2\lambda_1\lambda_2+\lambda_2\mu_1-\alpha_0^B\lambda_1\lambda_2} \\ 0 & \frac{z}{\mu_2}(z\lambda_2+\lambda_1 G_{\alpha^B}(z)-\alpha_0^B\lambda_1) \\ & +\frac{\lambda_1(\lambda_1+\lambda_2)(G_{\alpha^B}(z)-\alpha_0^B-z\alpha_1^B)}{\lambda_1^2+\lambda_2^2-\alpha_0^B\lambda_1^2+2\lambda_1\lambda_2+\lambda_2\mu_1-\alpha_0^B\lambda_1\lambda_2} - 1 \end{bmatrix} G_{\mathscr{A}} - z(\Psi_{10}^{-1}-\Psi_{01})G_{\mathscr{A}} + zI_{2\times 2}\right).
\end{aligned}$$

We know, $(zI_{2\times 2} - G_{\mathscr{A}})^{-1} = \frac{\begin{bmatrix} G_{\alpha^{11}}(z)-z & -G_{\alpha^{01}}(z) \\ -G_{\alpha^{10}}(z) & G_{\alpha^{00}}(z)-z \end{bmatrix}}{zG_{\alpha^{00}}(z)+zG_{\alpha^{11}}(z)-G_{\alpha^{00}}(z)G_{\alpha^{11}}(z)+G_{\alpha^{01}}(z)G_{\alpha^{10}}(z)-z^2}$, so we have:

$$\mathscr{D}(z) = \frac{\left\{\begin{array}{c} \begin{bmatrix} z^2\frac{\lambda_2}{\mu_2}-1 & \frac{\lambda_1^2(G_{\alpha^{BP}}(z)-\alpha_0^{BP}-z\alpha_1^{BP})}{\lambda_1^2+\lambda_2^2-\alpha_0^{BP}\lambda_1^2+2\lambda_1\lambda_2+\lambda_2\mu_1-\alpha_0^{BP}\lambda_1\lambda_2} \\ 0 & \frac{z}{\mu_2}(z\lambda_2+\lambda_1 G_{\alpha^{BP}}(z)-\alpha_0^{BP}\lambda_1) \\ & +\frac{\lambda_1(\lambda_1+\lambda_2)(G_{\alpha^{BP}}(z)-\alpha_0^{BP}-z\alpha_1^{BP})}{\lambda_1^2+\lambda_2^2-\alpha_0^{BP}\lambda_1^2+2\lambda_1\lambda_2+\lambda_2\mu_1-\alpha_0^{BP}\lambda_1\lambda_2}-1 \end{bmatrix}\mathscr{C}(z) - z(\Psi_{10}^{-1}-\Psi_{01})\mathscr{C}(z) \\ +z\begin{bmatrix} -(z-G_{\alpha^{11}}(z)) & -G_{\alpha^{01}}(z) \\ -G_{\alpha^{10}}(z) & -(z-G_{\alpha^{00}}(z)) \end{bmatrix} \end{array}\right\}}{zG_{\alpha^{00}}(z)+zG_{\alpha^{11}}(z)-G_{\alpha^{00}}(z)G_{\alpha^{11}}(z)+G_{\alpha^{01}}(z)G_{\alpha^{10}}(z)-z^2}, \tag{4.61}$$

in which

$$\mathscr{C}(z) = \begin{bmatrix} G_{\alpha^{00}}(z) & G_{\alpha^{01}}(z) \\ G_{\alpha^{10}}(z) & G_{\alpha^{11}}(z) \end{bmatrix}\begin{bmatrix} G_{\alpha^{11}}(z)-z & -G_{\alpha^{01}}(z) \\ -G_{\alpha^{10}}(z) & G_{\alpha^{00}}(z)-z \end{bmatrix}.$$

Ψ_{10}^{-1} can be calculated from (4.32) as $\Psi_{10}^{-1} = \frac{1}{\mu_2}\begin{bmatrix} \lambda_1+\lambda_2+\mu_2 & -\lambda_1 \\ -\mu_1 & \lambda_1+\lambda_2+\mu_1+\mu_2-\alpha_0^{BP}\lambda_1 \end{bmatrix}$, and $G_{\alpha^{L_k^1,L_{k+1}^1}}(z)$ is the GF of $\alpha^{L_k^1,L_{k+1}^1}$. It can be calculated from (4.3) as:

$$G_{\alpha^{L_k^1,L_{k+1}^1}}(z) = LT^{L_k^1,L_{k+1}^1}(\lambda_2-\lambda_2 z). \tag{4.62}$$

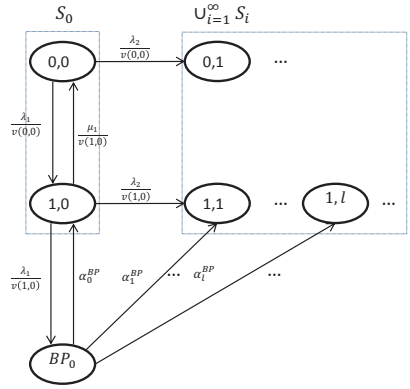

Figure 4.12: MC for the $c = 2$ servers case where $L_k^2 = 0$.

4.10.2 Transition Probabilities

The Transition Probabilities for $L_k^2 = 0$ when $c = 2$

As in Section 4.4.1, to find the one-step transition probabilities of the EMC, we first express the first-passage probability distribution from S_0 to $\cup_{i=1}^{\infty} S_i$.

We think of the MC after the k^{th} Class-2 departure as a MC with transient set: $S_0 \cup BP_0$, and absorbing sets: S_1 and $\cup_{i=2}^{\infty} S_i$. (Defining S_1 and $\cup_{i=2}^{\infty} S_i$ instead of $\cup_{i=1}^{\infty} S_i$ is for the computational convenience.) Let $\Gamma_{0\to 0}$, $\Gamma_{0\to 1}$ and $\Gamma_{0\to 2^+}$ be the one-step transition matrices from $S_0 \cup BP_0$ to $S_0 \cup BP_0$, S_1 and $\cup_{i=2}^{\infty} S_i$, respectively.

In Figure 4.12, we illustrate the arrival process of Class-2 jobs omitting details that are not relevant to the development of this case. From Figure 4.12, we get $\Gamma_{0\to 0}$, $\Gamma_{0\to 1}$ and $\Gamma_{0\to 2^+}$:

$$\Gamma_{0\to 0} = \begin{array}{c} \\ (0,0) \\ (1,0) \\ BP_0 \end{array} \begin{array}{c} \begin{array}{ccc} (0,0) & (1,0) & BP_0 \end{array} \\ \begin{bmatrix} 0 & \frac{\lambda_1}{v(0,0)} & 0 \\ \frac{\mu_1}{v(1,0)} & 0 & \frac{\lambda_1}{v(1,0)} \\ 0 & \alpha_0^{BP} & 0 \end{bmatrix} \end{array}, \quad \Gamma_{0\to 1} = \begin{array}{c} \\ (0,0) \\ (1,0) \\ BP_0 \end{array} \begin{array}{c} \begin{array}{cc} (0,1) & (1,1) \end{array} \\ \begin{bmatrix} \frac{\lambda_2}{v(0,0)} & 0 \\ 0 & \frac{\lambda_2}{v(1,0)} \\ 0 & \alpha_1^{BP} \end{bmatrix} \end{array},$$

$$\text{and } \Gamma_{0\to 2^+} = \begin{array}{c} \\ (0,0) \\ (1,0) \\ BP_0 \end{array} \begin{array}{c} \begin{array}{ccccc} (0,2) & (1,2) & (0,3) & (1,3) & \cdots \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & \cdots \\ 0 & 0 & 0 & 0 & \cdots \\ 0 & \alpha_2^{BP} & 0 & \alpha_3^{BP} & \cdots \end{bmatrix} \end{array}.$$

Let Ψ_{01} be the absorbing distribution matrix from S_0 to S_1. Let Ψ_{02} be the absorbing distribution

matrix from S_0 to $\cup_{i=2}^{\infty} S_i$. Using Lemma 2, we calculate Ψ_{01} and Ψ_{02} as:

$$\Psi_{01} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \cdot (I_{3\times 3} - \Gamma_{0\to 0})^{-1}\Gamma_{0\to 1} = \frac{\begin{bmatrix} \lambda_2(\lambda_1+\lambda_2+\mu_1-\alpha_0^{BP}\lambda_1) & \lambda_1(\lambda_2+\alpha_1^{BP}\lambda_1) \\ \lambda_2\mu_1 & (\lambda_1+\lambda_2)(\lambda_2+\alpha_1^{BP}\lambda_1) \end{bmatrix}}{\lambda_1^2+\lambda_2^2-\alpha_0^{BP}\lambda_1^2+2\lambda_1\lambda_2+\lambda_2\mu_1-\alpha_0^{BP}\lambda_1\lambda_2}, \tag{4.63}$$

and

$$\Psi_{02} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \cdot (I_{3\times 3} - \Gamma_{0\to 0})^{-1}\Gamma_{0\to 2^+}. \tag{4.64}$$

When the MC goes to $\cup_{i=1}^{\infty} S_i$, there are one or more Class-2 jobs in the system and there are no transitions in the EMC. As in Section 4.4.1, we use conditional probability to calculate transition probabilities of the EMC:

$$m_{(L_k^1,0)\to(L_{k+1}^1,L_{k+1}^2)} = \sum_{(q_1,q_2)\in\cup_{i=1}^{L_{k+1}^2+1} S_i} m_{(q_1,q_2)\to(L_{k+1}^1,L_{k+1}^2)} P\left\{(q_1,q_2) \mid (L_k^1,0)\right\}, \tag{4.65}$$

in which $m_{(q_1,q_2)\to(L_{k+1}^1,L_{k+1}^2)}$ is given in (4.34) and $P\left\{(q_1,q_2) \mid (L_k^1,0)\right\}$ is the corresponding probability of absorption in S_1 or $\cup_{i=2}^{\infty} S_i$ given in (4.63) and (4.64) respectively. Similar to (4.34), we must have $q_2 \in \left[1, L_{k+1}^2+1\right]$.

From (4.65), we get the matrices $M_{0\to L_{k+1}^2}$ in (4.36) for $L_{k+1}^2 \geq 0$.

The Transition Probabilities for $L_k^2 \geq c$ when $c = 2$

As in Section 4.5.1, we first think of the MC after the k^{th} Class-2 departure as a MC with transient set: $S_{L_k^2} \cup BP_{L_k^2}$, and absorbing sets: $S_{L_k^2-1}$ and $\cup_{i=L_k^2+1}^{\infty} S_i$. Let $\Gamma_{2\to 2}$, $\Gamma_{2\to 1}$ and $\Gamma_{2\to 3^+}$ be the one-step transition matrices from $S_{L_k^2} \cup BP_{L_k^2}$ to $S_{L_k^2} \cup BP_{L_k^2}$, $S_{L_k^2-1}$ and $\cup_{i=L_k^2+1}^{\infty} S_i$, respectively. From Figure

4.5, we get $\Gamma_{2\to 2}$, $\Gamma_{2\to 1}$ and $\Gamma_{2\to 3^+}$:

$$\Gamma_{2\to 2} = \begin{array}{c} \\ (0,L_k^2) \\ (1,L_k^2) \\ BP_{L_k^2} \end{array} \begin{array}{c} \begin{array}{ccc} (0,L_k^2) & (1,L_k^2) & BP_{L_k^2} \end{array} \\ \begin{bmatrix} 0 & \frac{\lambda_1}{v(0,L_k^2)} & 0 \\ \frac{\mu_1}{v(1,L_k^2)} & 0 & \frac{\lambda_1}{v(1,L_k^2)} \\ 0 & \alpha_0^{BP} & 0 \end{bmatrix} \end{array}, \quad \Gamma_{2\to 1} = \begin{array}{c} \\ (0,L_k^2) \\ (1,L_k^2) \\ BP_{L_k^2} \end{array} \begin{array}{c} \begin{array}{cc} (0,L_k^2-1) & (1,L_k^2-1) \end{array} \\ \begin{bmatrix} \frac{2\mu_2}{v(0,L_k^2)} & 0 \\ 0 & \frac{\mu_2}{v(1,L_k^2)} \\ 0 & 0 \end{bmatrix} \end{array},$$

$$\text{and } \Gamma_{2\to 3^+} = \begin{array}{c} \\ (0,L_k^2) \\ (1,L_k^2) \\ BP_{L_k^2} \end{array} \begin{array}{c} \begin{array}{ccccc} (0,L_k^2+1) & (1,L_k^2+1) & (0,L_k^2+2) & (1,L_k^2+2) & \cdots \end{array} \\ \begin{bmatrix} \frac{\lambda_2}{v(0,L_k^2)} & 0 & 0 & 0 & \cdots \\ 0 & \frac{\lambda_2}{v(1,L_k^2)} & 0 & 0 & \cdots \\ 0 & \alpha_1^{BP} & 0 & \alpha_2^{BP} & \cdots \end{bmatrix} \end{array}.$$

Then, with similar reasoning as in Section 4.5.1, we calculate $\mathscr{A}_i$ from:

$$\mathscr{A}_i = \begin{cases} \Psi_{21} \text{ for } i = 0 \\ \Psi_{23^+} \begin{bmatrix} \mathscr{A}_{i-1}^T & \cdots & \mathscr{A}_1^T & \mathscr{A}_0^T & \mathbf{0}_{2\times\infty} \end{bmatrix}^T \text{ for } i \geq 1 \end{cases}, \tag{4.66}$$

where

$$\Psi_{21} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \cdot (I - \Gamma_{2\to 2})^{-1}\Gamma_{2\to 1} = \frac{\begin{bmatrix} 2\mu_2(\lambda_1+\lambda_2+\mu_1+\mu_2-\alpha_0^{BP}\lambda_1) & \lambda_1\mu_2 \\ 2\mu_1\mu_2 & \mu_2(\lambda_1+\lambda_2+2\mu_2) \end{bmatrix}}{(\lambda_1+\lambda_2+\mu_1+\mu_2-\alpha_0^{BP}\lambda_1)(\lambda_1+\lambda_2+2\mu_2)-\lambda_1\mu_1}. \tag{4.67}$$

and

$$\Psi_{23^+} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \cdot (I - \Gamma_{2\to 2})^{-1}\Gamma_{2\to 3^+}. \tag{4.68}$$

Notice that $\mathscr{A}_i$ only depends on $\mathscr{A}_0, \mathscr{A}_1, \ldots, \mathscr{A}_{i-1}$. Thus, $\mathscr{A}_i$ can be calculated recursively from $\mathscr{A}_0$ (which is Ψ_{21} in (4.67)).

4.10.3 Proofs

Proof of Lemma 2

The one step transition probability of the MC can be written in matrix form as

$$P = \begin{array}{c} \\ T \\ A \end{array} \begin{array}{c} \begin{array}{cc} T & A \end{array} \\ \begin{bmatrix} \Gamma_{T\to T} & \Gamma_{T\to A} \\ 0 & I \end{bmatrix} \end{array},$$

where I is the identity matrix. Then, P^n represents the n step transition probabilities for the MC. Using induction, we obtain

$$P^n = \begin{bmatrix} \Gamma_{T\to T}^n & \sum_{i=0}^n \Gamma_{T\to T}^i \Gamma_{T\to A} \\ 0 & I \end{bmatrix}.$$

By letting n go to infinity and noting that $\sum_{i=0}^n \Gamma_{T\to T}^i = (I - \Gamma_{T\to T})^{-1}$, the probability that the system eventually reaches a state $A_i \in A$ is as given in the Lemma.

Proof of Lemma 3

After some algebra, we can write $(4.47 - 4.49)$ as:

$$B_1 = \frac{(\lambda_1 + 2\mu_2 + \lambda_2 B_0)B_0}{\lambda_1 + \lambda_2 B_0}, \tag{4.69}$$

$$B_2 = \frac{1}{\lambda_1}((\lambda_1 + \lambda_2 + \mu_1 + \mu_2 - \lambda_2(B_1 - B_0))B_1 - (\lambda_2 + \mu_1)B_0 - \mu_2), \tag{4.70}$$

$$B_{i+1} = \frac{1}{\lambda_1}((\lambda_1 + \lambda_2 + 2\mu_1 - \lambda_2(B_1 - B_0))B_i - 2\mu_1 B_{i-1} - \lambda_2 B_0) \text{ for } i \geq 2. \tag{4.71}$$

Let $\Delta_i^B = B_{i+1} - B_i$ for $i \geq 0$, be the step difference of the sequence B_i. So, we have

$$B_i = B_1 + \sum_{j=1}^{i-1} \Delta_j^B \text{ for } i \geq 2. \tag{4.72}$$

From the definition of Δ_i^B and (4.69), we get $\Delta_0^B = \frac{2\mu_2 B_0}{\lambda_1 + \lambda_2 B_0}$. Similarly, we get from $(4.69 - 4.71)$

$$\Delta_1^B = \frac{1}{\lambda_1}((\lambda_1 + \lambda_2 + \mu_1 + \mu_2 - \lambda_2\Delta_0^B)\Delta_0^B - \frac{\lambda_1\mu_2\Delta_0^B}{2\mu_2 - \lambda_2\Delta_0^B} - \mu_2), \tag{4.73}$$

$$\Delta_2^B = \frac{1}{\lambda_1}((\lambda_1 + \lambda_2 + 2\mu_1 - \lambda_2\Delta_0^B)\Delta_1^B - (\mu_1 + \mu_2)\Delta_0^B - \frac{\lambda_1\mu_2\Delta_0^B}{2\mu_2 - \lambda_2\Delta_0^B} + \mu_2), \tag{4.74}$$

$$\Delta_i^B = \frac{(\lambda_1 + \lambda_2 + 2\mu_1 - \lambda_2\Delta_0^B)}{\lambda_1}\Delta_{i-1}^B - \frac{2\mu_1}{\lambda_1}\Delta_{i-2}^B \text{ for } i \geq 3. \tag{4.75}$$

We notice that Δ_i^B is a linear homogeneous function of Δ_{i-1}^B and Δ_{i-2}^B , so Δ_i^B is a *linear homogeneous recurrence sequence* (see e.g., Green and Knuth (1990) Chapter 2). The solution to the recurrence sequence takes the form $\Delta_i^B = \kappa_1 g_1^i + \kappa_2 g_2^i$, $i \geq 1$, where g_1 and g_2 are roots of the *Characteristic Polynomial*: $CP(g) = \lambda_1 g^2 - (\lambda_1 + \lambda_2 + 2\mu_1 - \lambda_2\Delta_0^B)g + 2\mu_1$. Note that because $B_i \in [0, 1]$, we have

$$\lim_{i\to\infty} \Delta_i^B = 0. \tag{4.76}$$

For Δ_i^B to satisfy (4.76), i.e., converge to zero, either $g_j < 1$ or $\kappa_j = 0$ for both $j = 1, 2$.

Because B_0,$B_1 \in [0,1]$, we have $\Delta_0^B < 1$, so we have $CP(1) = \lambda_2(\Delta_0^B - 1) < 0$. Thus, $CP(g)$ has only one root that is smaller than one:

$$g = \frac{1}{2\lambda_1}(\lambda_1 + \lambda_2 + 2\mu_1 - \lambda_2\Delta_0^B - \sqrt{(\lambda_1 + \lambda_2 + 2\mu_1 - \lambda_2\Delta_0^B)^2 - 8\lambda_1\mu_1}). \tag{4.77}$$

(It is also easy to verify that g is greater than zero.) For the other root that is greater than one, the corresponding κ_j must be zero. Thus, Δ_i^B takes the form

$$\Delta_i^B = \kappa g^i,\ i \geq 1. \tag{4.78}$$

Notice that g is a function of Δ_0^B, so in the expression of Δ_i^B we have two unknowns: κ and Δ_0^B. Substituting (4.78) into (4.73) and (4.74) gives

$$\kappa g = \frac{1}{\lambda_1}((\lambda_1 + \lambda_2 + \mu_1 + \mu_2 - \lambda_2\Delta_0^B)\Delta_0^B - \frac{\lambda_1\mu_2\Delta_0^B}{2\mu_2 - \lambda_2\Delta_0^B} - \mu_2), \tag{4.79}$$

$$\kappa g^2 = \frac{1}{\lambda_1}((\lambda_1 + \lambda_2 + 2\mu_1 - \lambda_2\Delta_0^B)\Delta_1^B - (\mu_1 + \mu_2)\Delta_0^B - \frac{\lambda_1\mu_2\Delta_0^B}{2\mu_2 - \lambda_2\Delta_0^B} + \mu_2). \tag{4.80}$$

Dividing (4.80) with (4.79) gives:

$$g = -\frac{1}{\lambda_1}\frac{\begin{array}{c}\lambda_2^3{\Delta_0^B}^4 - \lambda_2^2(2\lambda_1 + 2\lambda_2 + 3\mu_1 + 3\mu_2){\Delta_0^B}^3 \\ +\lambda_2(\lambda_1^2 + 2\lambda_1\lambda_2 + \lambda_2^2 + 2\lambda_1\mu_1 + 3\lambda_1\mu_2 + 3\lambda_2\mu_1 + 6\lambda_2\mu_2 + 2\mu_1^2 + 8\mu_1\mu_2 + 2\mu_2^2){\Delta_0^B}^2 \\ -\mu_2(3\lambda_2^2 + 8\lambda_2\mu_1 + 4\lambda_2\mu_2 + 3\lambda_1\lambda_2 + 4\mu_1^2 + 4\mu_1\mu_2 + 2\lambda_1\mu_1)\Delta_0^B + 2\mu_2^2(\lambda_2 + 2\mu_1)\end{array}}{\lambda_2^2{\Delta_0^B}^3 - \lambda_2(\lambda_2 + \lambda_1 + \mu_1 + 3\mu_2){\Delta_0^B}^2 + \mu_2(2\mu_2 + \lambda_1 + 3\lambda_2 + 2\mu_1)\Delta_0^B - 2\mu_2^2}. \tag{4.81}$$

Substituting $\Delta_0^B = -\frac{\lambda_1 g^2 - (\lambda_1+\lambda_2+2\mu_1)g+2\mu_1}{\lambda_2 g}$ into (4.81) gives a polynomial equation of degree six: $0 = \lambda_1^3(\mu_1 - \mu_2)g^6 - \lambda_1^2(6\mu_1^2 + 2\mu_2^2 + 2\lambda_1\mu_1 - \lambda_1\mu_2 + 2\lambda_2\mu_1 - \lambda_2\mu_2 - 8\mu_1\mu_2)g^5 + (\lambda_1^3\mu_1 + 2\lambda_1^2\lambda_2\mu_1 + 16\lambda_1^2\mu_1^2 - 14\lambda_1^2\mu_1\mu_2 + 2\lambda_1^2\mu_2^2 + \lambda_1\lambda_2^2\mu_1 + 8\lambda_1\lambda_2\mu_1^2 - 6\lambda_1\lambda_2\mu_1\mu_2 + 12\lambda_1\mu_1^3 - 20\lambda_1\mu_1^2\mu_2 + 8\lambda_1\mu_1\mu_2^2)g^4 - (14\lambda_1^2\mu_1^2 - 6\lambda_1^2\mu_1\mu_2 + 16\lambda_1\lambda_2\mu_1^2 - 6\lambda_1\lambda_2\mu_1\mu_2 + 40\lambda_1\mu_1^3 - 44\lambda_1\mu_1^2\mu_2 - 8\lambda_1\mu_1\mu_2^2 + 2\lambda_2^2\mu_1^2 + 8\lambda_2\mu_1^3 - 8\lambda_2\mu_1^2\mu_2 + 8\mu_1^4 - 16\mu_1^3\mu_2 + 8\mu_1^2\mu_2^2)g^3 + 4\mu_1^2(\lambda_1^2 + 2\lambda_1\lambda_2 + 11\lambda_1\mu_1 - 6\lambda_1\mu_2 + \lambda_2^2 + 6\lambda_2\mu_1 - 3\lambda_2\mu_2 + 8\mu_1^2 - 10\mu_1\mu_2 + 2\mu_2^2)g^2 - (40\mu_1^4 + 16\lambda_1\mu_1^3 + 16\lambda_2\mu_1^3 - 24\mu_1^3\mu_2)g + 16\mu_1^4$

If $\mu_1 \neq \mu_2$, then we have several possible solutions:

$$\dot{g} = \frac{\mu_1}{2\lambda_1(\mu_1 - \mu_2)}(\lambda_1 + \lambda_2 + 2\mu_1 - 2\mu_2 - \sqrt{(\lambda_1 + \lambda_2 + 2\mu_1 - 2\mu_2)^2 - 8\lambda_1(\mu_1 - \mu_2)}),$$

$$\ddot{g} = \frac{\mu_1}{2\lambda_1(\mu_1 - \mu_2)}(\lambda_1 + \lambda_2 + 2\mu_1 - 2\mu_2 + \sqrt{(\lambda_1 + \lambda_2 + 2\mu_1 - 2\mu_2)^2 - 8\lambda_1(\mu_1 - \mu_2)}),$$

and roots of $O(g) = a_4x^4 + a_3x^3 + a_2x^2 + a_1x + a_0$, where $a_4 = \lambda_1^2,\ a_3 = \lambda_1(2\mu_2 - \lambda_2 - 4\mu_1 - \lambda_1),\ a_2 =$

$2(2\mu_1^2+4\lambda_1\mu_1-\lambda_1\mu_2+\lambda_2\mu_1-2\mu_1\mu_2), a_1=4\mu_1(\mu_2-\lambda_1-\lambda_2-3\mu_1),\ a_0=8\mu_1^2$. It is easy to check that $\dot{g}>1$; if $\mu_1>\mu_2$, then $\ddot{g}>\dot{g}$ so $\ddot{g}>1$; if $\mu_1<\mu_2$, then $\ddot{g}<0$. So, g cannot be $\dot{g}$ or $\ddot{g}$, and g must be one of the four roots of $O(g)$.

The four roots of a quartic function are well known. Let $\Delta_1=a_2^2-3a_3a_1+12a_4a_0$, $\Delta_2=2a_2^3-9a_3a_2a_1+27a_4a_1^2+27a_3^2a_0-72a_4a_2a_0$, and $\Delta=\frac{\sqrt[3]{2}\Delta_1}{3a_4\sqrt[3]{\Delta_2+\sqrt{-4\Delta_1^3+\Delta_2^2}}}+\frac{\sqrt[3]{\Delta_2+\sqrt{-4\Delta_1^3+\Delta_2^2}}}{3\sqrt[3]{2}a_4}$, then the four roots of $O(g)$ are

$$x_1 = -\frac{a_3}{4a_4}-\frac{1}{2}\sqrt{\frac{a_3^2}{4a_4^2}-\frac{2a_2}{3a_4}+\Delta}-\frac{1}{2}\sqrt{\frac{a_3^2}{2a_4^2}-\frac{4a_2}{3a_4}-\Delta-\frac{-\frac{a_3^3}{a_4^3}+\frac{4a_3a_2}{a_4^2}-\frac{8a_1}{a_4}}{4\sqrt{\frac{a_3^2}{4a_4^2}-\frac{2a_2}{3a_4}+\Delta}}}, \tag{4.82}$$

$$x_2 = -\frac{a_3}{4a_4}-\frac{1}{2}\sqrt{\frac{a_3^2}{4a_4^2}-\frac{2a_2}{3a_4}+\Delta}+\frac{1}{2}\sqrt{\frac{a_3^2}{2a_4^2}-\frac{4a_2}{3a_4}-\Delta-\frac{-\frac{a_3^3}{a_4^3}+\frac{4a_3a_2}{a_4^2}-\frac{8a_1}{a_4}}{4\sqrt{\frac{a_3^2}{4a_4^2}-\frac{2a_2}{3a_4}+\Delta}}}, \tag{4.83}$$

$$x_3 = -\frac{a_3}{4a_4}+\frac{1}{2}\sqrt{\frac{a_3^2}{4a_4^2}-\frac{2a_2}{3a_4}+\Delta}-\frac{1}{2}\sqrt{\frac{a_3^2}{2a_4^2}-\frac{4a_2}{3a_4}-\Delta+\frac{-\frac{a_3^3}{a_4^3}+\frac{4a_3a_2}{a_4^2}-\frac{8a_1}{a_4}}{4\sqrt{\frac{a_3^2}{4a_4^2}-\frac{2a_2}{3a_4}+\Delta}}}, \tag{4.84}$$

$$x_4 = -\frac{a_3}{4a_4}+\frac{1}{2}\sqrt{\frac{a_3^2}{4a_4^2}-\frac{2a_2}{3a_4}+\Delta}+\frac{1}{2}\sqrt{\frac{a_3^2}{2a_4^2}-\frac{4a_2}{3a_4}-\Delta+\frac{-\frac{a_3^3}{a_4^3}+\frac{4a_3a_2}{a_4^2}-\frac{8a_1}{a_4}}{4\sqrt{\frac{a_3^2}{4a_4^2}-\frac{2a_2}{3a_4}+\Delta}}}. \tag{4.85}$$

Because $O(1)<0$ and $\lim_{g\to\infty}O(g)=\infty$, $O(g)$ has at least one root in $(1,\infty)$. Because $O(0)=8\mu_1^2>0$ and $O(1)=-\lambda_2(\lambda_1+2\mu_1)<0$, $O(g)$ has at least one root in $(0,1)$. Because $O(0)=8\mu_1^2>0$ and $\lim_{g\to-\infty}O(g)=\infty$, $O(g)$ has either two or no roots in $(-\infty,0)$. Next, we prove $O(g)$ has only one root in $(0,1)$.

From $\sum_{i=1}^{2}\frac{\lambda_i}{\mu_i}<2$, we get that $\mu_2>\frac{\lambda_2\mu_1}{2\mu_1-\lambda_1}$. Then we discuss the following three cases:

1. If $\frac{\lambda_2\mu_1}{2\mu_1-\lambda_1}\geq\frac{\lambda_2+\lambda_1+4\mu_1}{2}$, then $\mu_2>\frac{\lambda_2+\lambda_1+4\mu_1}{2}$, i.e., $a_3=\lambda_1(2\mu_2-\lambda_2-4\mu_1-\lambda_1)>0$. Note from (4.82) that, in this case, x_1 is either a complex root or a negative real root:

   (a) If x_1 is a complex root, because of the *Complex Conjugate Root Theorem* (i.e., Jeffrey 2005), x_2 must be the other complex root. Obviously $x_4\geq x_3$, so we know $x_4\in(1,\infty)$ and $x_3\in(0,1)$.

   (b) If x_1 is a negative real root, because $O(g)$ has either two or no roots in $(-\infty,0)$, $O(g)$ must have two negative real roots. Therefore, $O(g)$ has only one root in $(0,1)$.

2. If $\frac{\lambda_2\mu_1}{2\mu_1-\lambda_1}<\frac{\lambda_2+\lambda_1+4\mu_1}{2}$ and $\mu_2>\frac{\lambda_2+\lambda_1+4\mu_1}{2}$, then as in the first case, we know $x_4\in(1,\infty)$ and $x_3\in(0,1)$.

3. If $\frac{\lambda_2\mu_1}{2\mu_1-\lambda_1}<\frac{\lambda_2+\lambda_1+4\mu_1}{2}$ and $\frac{\lambda_2\mu_1}{2\mu_1-\lambda_1}<\mu_2\leq\frac{\lambda_2+\lambda_1+4\mu_1}{2}$, we let $\epsilon=\frac{\lambda_2+\lambda_1+4\mu_1}{2}-\mu_2$ (i.e., $0\leq\epsilon<$

$\frac{\lambda_1^2+2\lambda_1\mu_1+\lambda_2\lambda_1-8\mu_1^2}{2(\lambda_1-2\mu_1)}$), $O_1(g) = -2\lambda_1 g^3 + 2(\lambda_1+2\mu_1)g^2 - 4\mu_1 g$ and $O_2(g) = \lambda_1^2 g^4 + (-\lambda_1^2 + 2\lambda_1\mu_1 - \lambda_2\lambda_1 - 4\mu_1^2)g^2 - 2\mu_1(2\mu_1+\lambda_1+\lambda_2)g + 8\mu_1^2$, so that $O(g) = \epsilon R_1(g) + O_2(g)$.

$O_1(g)$ and $O_2(g)$ have some properties that are easy to derive that can be used to identify the root we want.

- $O_1(g)$ is a convex function on $[0,1]$ and $O_1(0) = O_1(1) = 0$.
- $O_2(g)$ is a decreasing function on $[0,1]$, $O_2(0) = 8\mu_1^2 > 0$ and $O_2(1) = -\lambda_2(\lambda_1+2\mu_1) < 0$.
  To prove $O_2(g)$ is a decreasing function on $[0,1]$, we just need to prove the first derivative of $O_2(g)$ is negative, i.e., $O_2'(g) = 4\lambda_1^2 g^3 + (4\lambda_1\mu_1 - 2\lambda_1^2 - 2\lambda_2\lambda_1 - 8\mu_1^2)g - (4\mu_1^2 + 2\lambda_1\mu_1 + 2\lambda_2\mu_1) < 0$, for $\forall g \in [0,1]$.
  Obviously, $O_2'(0) = -(4\mu_1^2 + 2\lambda_1\mu_1 + 2\lambda_2\mu_1) < 0$ and $O_2'(1) = -2(4\mu_1^2 - \lambda_1^2) - 2\mu_1(2\mu_1 - \lambda_1) - 2\lambda_2\lambda_1 - 2\lambda_2\mu_1 < 0$. We know the second derivative of $O_2(g)$ is $O_2''(g) = 12\lambda_1^2 g^2 + (4\lambda_1\mu_1 - 2\lambda_1^2 - 2\lambda_2\lambda_1 - 8\mu_1^2)$ and $O_2''(0) = -4\mu_1(2\mu_1 - \lambda_1) - 2\lambda_1^2 - 2\lambda_2\lambda_1 < 0$. If there exists a point $\bar{g}$ in $[0,1]$ such that $O_2'(\bar{g}) > 0$, then $O_2'(g)$ must have two critical points in $[0,1]$, i.e., $O_2''(g)$ must have two roots in $[0,1]$. However, we know $O_2''(g)$ has one negative root and one positive root. Therefore, $O_2'(g) < 0$ for $\forall g \in [0,1]$. Then, we know $O_2(g)$ is a decreasing function for $\forall g \in [0,1]$.

It seems obvious that for $\forall \epsilon \geq 0$, $O(g) = \epsilon R_1(g) + O_2(g)$ has only one root in $(0,1)$.

Hence, we proved $O(g)$ has only one root in $(0,1)$. Then, we just need to pick up the root in $(0,1)$ from the four roots of $O(g)$, which is not difficult. Once we get g, solving (4.77) and (4.79) gives the corresponding Δ_0^B and κ as given in Lemma 3.

Proof of Lemma 4

As in the Proof of Lemma 3, we write $(4.44 - 4.46)$ in another form:

$$F_1 = \frac{(\lambda_1+\mu_2+\lambda_2 B_0)F_0}{\lambda_1+\lambda_2 B_0} = \frac{2\lambda_1+2\mu_2-\lambda_2\Delta_0^B}{2\lambda_1}F_0, \tag{4.86}$$

$$F_2 = \frac{1}{\lambda_1}((\lambda_1+\lambda_2+\mu_1+\mu_2)F_1 - (\lambda_2+\mu_1)F_0 - \lambda_2 B_1(F_1-F_0) - \mu_2), \tag{4.87}$$

$$F_{i+1} = \frac{1}{\lambda_1}((\lambda_1+\lambda_2+2\mu_1)F_i - 2\mu_1 F_{i-1} - \lambda_2 B_i(F_1-F_0) - \lambda_2 F_0) \text{ for } i \geq 2. \tag{4.88}$$

Let $\Delta_i^F = F_{i+1} - F_i$ be the step difference of the sequence F_i. So, we have

$$F_i = F_1 + \sum_{j=1}^{i-1} \Delta_j^F \text{ for } i \geq 2.$$

Because $F_i \in [0,1]$, we have $\lim_{i\to\infty} \Delta_i^F = 0$. Using (4.86), we get $\Delta_0^F = \frac{\mu_2 F_0}{\lambda_1+\lambda_2 B_0}$. Similarly, from $(4.86 - 4.88)$, we get

$$\begin{aligned}
\Delta_1^F &= \frac{1}{\lambda_1}((\lambda_1+\lambda_2+\mu_1+\mu_2-\lambda_2\Delta_0^B)\Delta_0^F-\mu_2),\\
\Delta_2^F &= \frac{1}{\lambda_1}((\lambda_1+\lambda_2+2\mu_1)\Delta_1^F-(\lambda_2\kappa g+\mu_1+\mu_2)\Delta_0^F-(\lambda_1+\lambda_2 B_0)\Delta_0^F+\mu_2),\\
\Delta_i^F &= \frac{(\lambda_1+\lambda_2+2\mu_1)}{\lambda_1}\Delta_{i-1}^F-\frac{2\mu_1}{\lambda_1}\Delta_{i-2}^F-\frac{\lambda_2\kappa\Delta_0^F}{\lambda_1 g}g^i \text{ for } i\geq 3.
\end{aligned}$$

Note that Δ_i^F is a linear non-homogeneous function of Δ_{i-1}^F and Δ_{i-2}^F, so Δ_i^F is a *non-homogeneous recurrence sequence* (see e.g., Green and Knuth (1990) Chapter 2), with solution of the form

$$\Delta_i^F = \xi_1 h_1^i + \xi_2 h_2^i + \xi_3 g^i,$$

where g is given in Lemma 3; h_1 and h_2 are roots of $\lambda_1 h^2 - (\lambda_1+\lambda_2+2\mu_1)h + 2\mu_1 = 0$. We know one of the two roots is greater than one. Because Δ_i^F converges to zero, with the same discussion in the proof of Lemma 3, we get that Δ_i^F has the form:

$$\Delta_i^F = \xi_1 h^i + \xi_2 g^i,\ i \geq 1$$

where $h = \frac{1}{2\lambda_1}((\lambda_1+\lambda_2+2\mu_1) - \sqrt{(\lambda_1+\lambda_2+2\mu_1)^2 - 8\lambda_1\mu_1})$. To find ξ_1, ξ_2 and Δ_0^F, we solve three equations

$$\Delta_1^F = \xi_1 h + \xi_2 g,\ \Delta_2^F = \xi_1 h^2 + \xi_2 g^2,\ \Delta_3^F = \xi_1 h^3 + \xi_2 g^3.$$

Notice that $\Delta_i^F, i = 1,2,3$ are all linear functions of Δ_0^F, so it is not hard to get the expression for ξ_1, ξ_2 and Δ_0^F in Lemma 4.

Proof of Lemma 5

Subtracting the $(i-1)^{st}$ equation from the i^{th} equation given in (4.52-4.53) yields

$$2\mu_1\Pi_i = (\lambda_1+\lambda_2+2\mu_1)\Pi_{i-1} - \lambda_1\Pi_{i-2} \text{ for } i \geq 3.$$

This means that Π_i is a linear homogeneous recurrence sequence. The solution to the recurrence sequence takes the form

$$\Pi_i = \omega_1 f_1^i + \omega_2 f_2^i,\ i \geq 1,$$

where ω_1 and ω_2 are roots of

$$2\mu_1 f^2 - (\lambda_1 + \lambda_2 + 2\mu_1)f + \lambda_1 = 0. \tag{4.89}$$

Because $\Pi_i \in [0,1]$, we know either $f_j < 1$ or $\omega_j = 0$ for both $j = 1, 2$. Equation (4.89) has one root greater than one and the other root smaller than one. For the root greater than one, the corresponding ω_j must be zero. Thus, Π_i takes the form

$$\Pi_i = \omega f^i \text{ for } i \geq 1, \tag{4.90}$$

where $f = \frac{1}{4\mu_1}(\lambda_1 + \lambda_2 + 2\mu_1 - \sqrt{(\lambda_1 + \lambda_2 + 2\mu_1)^2 - 8\lambda_1\mu_1})$, which is the root smaller than one. Substituting Π_1 in (4.90) gives $\omega = \frac{\Pi_1}{f}$. From (4.51), we get

$$\Pi_1 = \frac{1}{\mu_1}((\lambda_1 + \lambda_2)\Pi_0 - \lambda_2(1 - \varphi_1)).$$

Therefore, from

$$1 = \sum_{i=0}^{\infty} \Pi_i = \frac{\Pi_1}{1-f} + \Pi_0 = \frac{(\lambda_1 + \lambda_2)\Pi_0 - \lambda_2(1-\varphi_1)}{\mu_1(1-f)} + \Pi_0$$

we get $\Pi_0 = \frac{\mu_1(1-f)+\lambda_2(1-\varphi_1)}{\mu_1(1-f)+(\lambda_1+\lambda_2)}$. Therefore, Π_i can be expressed as a function of φ_1 as in (4.54).

4.10.4 Algorithms

Algorithm 4 *Calculate the transition matrix of the EMC for* $\forall c \geq 2$.

Step 1*: Let* $\Gamma_{c\to c}$, $\Gamma_{c\to(c-1)}$ *and* $\Gamma_{c\to(c+1)^+}$ *be the one-step transition matrices from* $S_c \cup BP_c$ *to* $S_c \cup BP_c$, S_{c-1} *and* $\cup_{j=c+1}^{\infty} S_j$. *Set* $\Psi_{c1} = \begin{bmatrix} I_{c\times c} & \mathbf{0}_{c\times 1} \end{bmatrix} \cdot (I - \Gamma_{c\to c})^{-1}\Gamma_{c\to(c-1)}$ *and* $\Psi_{c2} = \begin{bmatrix} I_{c\times c} & \mathbf{0}_{c\times 1} \end{bmatrix} \cdot (I - \Gamma_{c\to c})^{-1}\Gamma_{c\to(c+1)^+}$. *Set* $\mathscr{A}_0 = \Psi_{c1}$ *and let* $i = 1$.

Step 2*: Set* $\mathscr{A}_i = \Psi_{c2} \begin{bmatrix} \mathscr{A}_{i-1}^T & \cdots & \mathscr{A}_1^T & \mathscr{A}_0^T & \mathbf{0}_{c\times\infty} \end{bmatrix}^T$.

Step 3*: Let* $i = i+1$. *If* $\max(\mathscr{A}_i) > Tolerance$, *then go to* ***Step 2****; else set* $Limit = i$ *and* $i = c-1$, *and go to* ***Step 4***

Step 4*: Let* $\Gamma_{i\to i}$, $\Gamma_{i\to(i-1)}$ *and* $\Gamma_{i\to(i+1)^+}$ *be the one-step transition matrices from* $S_i \cup BP_i$ *to* $S_i \cup BP_i$, S_{i-1} *and* $\cup_{j=i+1}^{\infty} S_j$. *Set* $\Psi_{i1} = \begin{bmatrix} I_{c\times c} & \mathbf{0}_{c\times 1} \end{bmatrix} \cdot (I - \Gamma_{i\to i})^{-1}\Gamma_{i\to(i-1)}$, *and* $\Psi_{i2} = \begin{bmatrix} I_{c\times c} & \mathbf{0}_{c\times 1} \end{bmatrix} \cdot (I - \Gamma_{i\to i})^{-1}\Gamma_{i\to(i+1)^+}$. *Let* $j = 0$.

Step 5*: If* $j < i-1$, *then set* $M_{i\to j} = \mathbf{0}_{c\times c}$; *else if* $j = i-1$, *then set* $M_{i\to j} = \Psi_{i1}$; *else if* $i \leq j < c-1$, *then set* $M_{i\to j} = \Psi_{i2} \begin{bmatrix} M_{i+1\to j}^T & \cdots & M_{c-2\to j}^T & M_{c-1\to j}^T & \mathbf{0}_{c\times\infty} \end{bmatrix}^T$; *else set* $M_{i\to j} = \Psi_{i2} \begin{bmatrix} M_{i+1\to j}^T & \cdots & M_{c-1\to j}^T & \mathscr{A}_{j-c+1}^T & \cdots & \mathscr{A}_0^T & \mathbf{0}_{c\times\infty} \end{bmatrix}^T$.

Step 6*: Let* $j = j+1$. *If* $j < Limit$, *then go to* ***Step 5****; else let* $i = i-1$. *If* $i \geq 1$, *then let* $j = 0$ *and go to* ***Step 5****; else let* $i = 0$ *and go to* ***Step 7***.

Step 7: *Let* $\Gamma_{0\to 0}$ *and* $\Gamma_{0\to 1^+}$ *be the one-step transition matrices from* $S_0 \cup BP_0$ *to* $S_0 \cup BP_0$, $\cup_{j=1}^{\infty} S_j$. *Set* $\Psi_0 = \begin{bmatrix} I_{c\times c} & \mathbf{0}_{c\times 1} \end{bmatrix} \cdot (I - \Gamma_{0\to 0})^{-1}\Gamma_{0\to 1^+}$. *Let* $j = 0$.

Step 8: *If* $0 \le j < c-1$, *then set* $M_{0\to j} = \Psi_0 \begin{bmatrix} M_{1\to j}^T & \cdots & M_{c-2\to j}^T & M_{c-1\to j}^T & \mathbf{0}_{c\times\infty} \end{bmatrix}^T$; *else if* $M_{0\to j} = \Psi_0 \begin{bmatrix} M_{i+1\to j}^T & \cdots & M_{c-1\to j}^T & \mathscr{A}_{j-c+1}^T & \cdots & \mathscr{A}_0^T & \mathbf{0}_{c\times\infty} \end{bmatrix}^T$.

Step 9: *Let* $j = j+1$. *If* $j < Limit$, *go to* ***Step 8***; *else set* $G = \mathscr{A}_0$ *and go to* ***Step 10***.

Step 10: *Set* $G = \sum_{i=0}^{Limit} \mathscr{A}_i G^i$.

Step 11: *If* $\max(G - \sum_{i=0}^{Limit} \mathscr{A}_i G^i) > Tolerance$, *then go to* ***Step 10***; *else set* $\hat{\mathbf{L}} = \begin{bmatrix} M_{0\to 0} & \cdots & M_{0\to c-1} \\ \vdots & \ddots & \vdots \\ M_{c-1\to 0} & \cdots & M_{c-1\to c-1} \end{bmatrix}$

$\hat{\mathbf{B}} = \begin{bmatrix} \mathbf{0}_{c\times c(c-1)} & \mathscr{A}_0 \end{bmatrix}$, $\hat{\mathbf{F}}^{(i)} = \begin{bmatrix} M_{0\to i} \\ \vdots \\ M_{c-1\to i} \end{bmatrix}$ *for* $i = c, \ldots, Limit$, $\mathbf{B} = \mathscr{A}_0$, $\mathbf{F}^{(0)} = \mathbf{L} = \mathscr{A}_1$, $\mathbf{F}^{(i)} = \mathscr{A}_{i+1}$ *for* $i \ge 1$, $\hat{\mathbf{S}}^{(i)} = \sum_{j=i}^{Limit} \hat{\mathbf{F}}^{(j)} G^{j-i}$ *for* $i \ge 1$ *and* $\mathbf{S}^{(i)} = \sum_{j=i}^{Limit} \mathbf{F}^{(j)} G^{j-i}$ *for* $i \ge 0$, *and go to* ***Step 12***.

Step 12: *Solve* $\begin{bmatrix} \pi_{1\times c^2}^{(0)} & \pi_{1\times c}^{(1)} & \pi_{1\times c}^{(*)} \end{bmatrix}$

$$\cdot \begin{bmatrix} \mathbf{1}_{c^2\times 1} & \hat{\mathbf{L}} & \hat{\mathbf{F}}^{(1)} - \sum_{j=3}^{Limit} \hat{\mathbf{S}}^{(j)} G & \sum_{j=2}^{Limit} \hat{\mathbf{F}}^{(j)} + \sum_{j=3}^{Limit} \hat{\mathbf{S}}^{(j)} G \\ \mathbf{1}_{c\times 1} & \hat{\mathbf{B}} & \mathbf{L} - \sum_{j=2}^{Limit} \mathbf{S}^{(j)} G & \sum_{j=1}^{Limit} \mathbf{F}^{(j)} + \sum_{j=2}^{Limit} \mathbf{S}^{(j)} G \\ \mathbf{1}_{c\times 1} & \mathbf{0}_{c\times c} & \mathbf{B} - \sum_{j=1}^{Limit} \mathbf{S}^{(j)} G & \sum_{j=1}^{Limit} \mathbf{F}^{(j)} + \mathbf{L} + \sum_{j=1}^{Limit} \mathbf{S}^{(j)} G \end{bmatrix} = \left[1, \mathbf{0}_{1\times(c^2+2c)}\right].$$

Step 13: *Set* $\hat{\mathbf{F}}_{[k,i]} = \sum_{j=i}^{Limit} j^k \hat{\mathbf{F}}^{(j)}$ *for* $i \ge 1$ *and* $k = 0$ *or* 1, $\mathbf{F}_{[k,i]} = \sum_{j=i}^{Limit} j^k \mathbf{F}^{(j)}$ *for* $i \ge 1$, $b^{[1]} = -\pi^{(0)} \sum_{j=1}^{Limit} (j+1)\hat{\mathbf{F}}^{(j)} - \pi^{(1)}(2\mathbf{L} + \sum_{j=1}^{Limit} (j+2)\mathbf{F}^{(j)}) - \pi^{(*)}(\mathbf{L} + \sum_{j=1}^{Limit} (j+1)\mathbf{F}^{(j)})$, *and* $c^{[1]} = -\pi^{(0)} \sum_{j=2}^{Limit} j\hat{\mathbf{F}}_{[0,j]}\mathbf{1}^T - \pi^{(1)} \sum_{j=1}^{Limit} (j+1)\mathbf{F}_{[0,j]}\mathbf{1}^T - \pi^{(*)} \sum_{j=1}^{Limit} j\mathbf{F}_{[0,j]}\mathbf{1}^T$.

Step 14: *Solve* $r^{[1]} \cdot \left[\mathbf{B} + \mathbf{L} + \sum_{j=1}^{Limit} \mathbf{F}^{(j)}, (\mathbf{F}_{[1,1]} - \mathbf{B})\mathbf{1}^T\right] = \left[b^{[1]}, c^{[1]}\right]$.

Step 15: *Let* $E\left[L^2\right] = \pi_{1\times c^2}^{(0)} \cdot \begin{bmatrix} \mathbf{0}_{1\times c} & \mathbf{1}_{1\times c} & \cdots & (\mathbf{c}-\mathbf{1})_{1\times c} \end{bmatrix}^T + \pi_{1\times c}^{(1)} \cdot \begin{bmatrix} \mathbf{c}_{1\times c} \end{bmatrix}^T + (r^{[1]} + (c-1)\pi_{1\times c}^{(*)}) \cdot \mathbf{1}^T$ *and* ***Stop***.