SCHEDULING IN HEALTHCARE

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Woo-Hyung Cho

August 2022

SCHEDULING IN HEALTHCARE

Woo-Hyung Cho, Ph.D.

Cornell University 2022

In today's rapidly evolving, technology-driven and data-rich environment, we are increasingly being offered new information with which to make decisions. This dissertation examines how these changes are reshaping operations in healthcare settings, and seeks to better understand their impact on the system as a whole.

We first consider the use of machine-learned predictions of patient risk for triage and prioritization. We model this as a learning-augmented online scheduling problem where we are given good, but imperfect, information about each arriving patient's urgency level in advance. In this formulation, we face the challenges of decision making under imperfect information, and of responding dynamically to prediction error as we observe better data in real time. We propose a simple online policy for minimizing the total urgency-weighted costs of delay across all patients, and show that this policy is in fact the best possible in certain stylized settings.

Our work in the area of scheduling has also prompted some theoretical questions. We present a new proof of correctness of the celebrated Shortest Processing Time rule using geometric insights and linear programming techniques.

Finally, we examine the impact of online appointment scheduling platforms that offer patients the ability to observe and choose physicians that best meet their needs. In particular, some patients value the flexibility of seeing readily available physicians while others prefer dedicated service by a primary care provider. We study the effects of added flexibility in a multi-server queueing framework and show that even a small number of flexible patients can greatly benefit overall system performance.

BIOGRAPHICAL SKETCH

Woo-Hyung Cho was born March 22, 1991 in Seoul, South Korea. In 2013, she graduated magna cum laude from Princeton University with a B.S.E. in Operations Research and Financial Engineering. After three years of working at KKR, she moved to Ithaca, NY in the summer of 2016 to begin her doctoral studies in Operations Research at Cornell University.

Dedicated to my parents.

ACKNOWLEDGEMENTS

My name, Woo-Hyung, roughly translates to *with good friends, everything will go well*. It's a fitting name for the path I've taken in life so far. I would not be where I am today if it weren't for the people I have had the fortune to meet along the way.

First and foremost, I would like to express my deepest gratitude to my advisors David Shmoys and Shane Henderson for their patient guidance and support over the years. Thank you for the countless hours spent helping me carve out my own path in research. I have grown so much as a researcher, as a teacher, and as a person all thanks to you.

I would also like to thank Itai Gurvich and Huseyin Topaloglu for serving on my committee and for providing invaluable feedback. Special thanks goes to my wonderful mentor Jamol Pender without whom one of the chapters in this dissertation would not have been possible.

To my fellow Ph.D. students in ORIE, thank you for shaping my academic experience; it's been inspiring to be on this journey together. I am particularly grateful to Shuang Tao with whom I have had the pleasure of collaborating. To Michael Choi, Venus Lo, Andrew Daw, Yilun Chen, and Vasilis Charisopoulos, thank you for the warm friendship, mentorship, and community. I am also indebted to my lovely officemates Sander Aarts, Raul Astudillo, Chamsi Hssaine, Alberto Vera, and Amy Zhang, who have all made Rhodes 288 a second home.

I owe at least as much to my friends outside of Cornell for their support and encouragement from afar. Thank you, Irene C., Jawon K., Ha Eun K., Yuan Chang L., Stephanie P., Dean W., Audrye W., and Peter Y. for being such staunch supporters. I am particularly grateful to Julia Y., not just for the countless hours of working together over silent Zoom, but also for always having my back. I

am also thankful to Minkwang J. for being my safe haven all these years and for somehow never failing to find the right words of comfort when I needed them the most.

Most importantly, I would like to thank my parents and my little sister Woo Jin, whose endless love and support made this endeavor possible. I am grateful to Woo Jin for grudgingly visiting me in Ithaca more times than she ever wanted. To my parents, thank you for cheering me on from halfway across the world, for always believing in me, and for giving me the strength to finish this document. This dissertation is dedicated to you.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

In today's rapidly evolving, technology-driven and data-rich environment, we are increasingly being offered new information with which to make decisions. This dissertation examines how these changes are reshaping operations in healthcare settings, and seeks to better understand their impact on the system as a whole.

In Chapter 2, we begin by exploring recent interest in deploying machine learning algorithms for diagnostic radiology. Advances in modern learning techniques have made it possible to detect abnormalities in medical images within minutes. While machine-assisted diagnoses cannot yet reliably replace the human reviews of images by a radiologist, they could inform prioritization rules for determining the order by which to review patient cases so that patients with time-sensitive conditions could benefit from early intervention.

We study this scenario by formulating it as a learning-augmented online scheduling problem. We are given information about each arriving patient's urgency level in advance, but these predictions are inevitably error-prone. In this formulation, we face the challenges of decision making under imperfect information, and of responding dynamically to prediction error as we observe better data in real time. We propose a simple online policy and show that this policy is in fact the best possible in certain stylized settings. We also demonstrate that our policy achieves the two desiderata of online algorithms with predictions: consistency (performance improvement with prediction accuracy) and robustness (protection against the worst case). We complement our theoretical findings with empirical evaluations of the policy under settings that more accurately reflect clinical scenarios in the real world.

Our work in the area of scheduling has also prompted some theoretical questions, which we explore in Chapter 3. The Shortest Processing Time rule is one of the oldest results in scheduling theory that finds an optimal solution to the problem of scheduling jobs on identical parallel machines to minimize average job completion times. We present a new proof of correctness of the Shortest Processing Time rule using geometric insights and linear programming techniques. An extended abstract of this work has appeared in the proceedings of the MAPSP 2022 conference in June 2022.

Finally, in Chapter 4, we examine the impact of online appointment scheduling platforms that offer patients the ability to observe and choose physicians that best meet their needs. The increasing popularity of platforms like ZocDoc has fundamentally changed how patients experience appointment scheduling and raises new questions about its impact to the system as a whole. In this chapter, we focus on a specific aspect of this phenomenon in which we assume that some patients value the flexibility of seeing readily available physicians while others prefer dedicated service by a primary care provider. We study these effects of added flexibility in a multi-server queueing model that captures the performance trade-off between patients valuing flexibility (join the shortest-of-d queues) and patients wanting dedicated service (join a specific queue). Our analyses indicate that even a small number of flexible patients can greatly benefit overall system performance.

Chapter 4 began as an end-of-semester final project for ORIE 6520: *A Random Walk Through Applied Probability* taught by Jamol Pender in Spring 2018. Collaborative efforts with fellow PhD student Shuang Tao resulted in a research paper shortly thereafter. This chapter also appeared in co-author Shuang Tao's PhD dissertation submitted to Cornell University in 2020.

This research was supported in part by NSF CMMI-2035086, NSF/FDA SIR IIS-1935809, NSF CCF-1740822, NSF DMS-1839346, and NSF CCF-1522054.

CHAPTER 2

SCHEDULING WITH PREDICTIONS

Modern machine learning algorithms have been tremendously successful in a variety of application domains, and healthcare is no exception. In recent years, we have seen significant interest in deploying these algorithms for diagnostic radiology, a branch of medicine that uses imaging techniques such as X-rays, ultrasounds, and Magnetic Resonance Imaging (MRI) to diagnose a patient. The idea is to use these images as inputs to machine learning algorithms, which would then search for patterns that imply the presence of an abnormality. Advances in pattern recognition techniques for image processing and computer vision have made it possible for machine learning algorithms to detect abnormal conditions in medical images within minutes, or even seconds. Because this is still a nascent area of research, these algorithmic, machine-assisted diagnoses cannot yet reliably replace the thorough, human reviews of images by a radiologist. Meanwhile, they could be used to prioritize and speed up the review of images that are flagged as likely to contain time-sensitive conditions.

To make this more concrete, imagine a group of patients who have had diagnostic images taken after a referral. Radiologists are tasked with processing each patient case, which typically consists of reading images, then communicating any findings by filing a radiology report and sending it back to the referring provider. Appropriate patient care and treatment begin only upon case completion at the radiology department, so it is in the best interest of the patient for radiologists to organize their workflow in a way that prioritizes cases by urgency. This is especially true for patients with time-sensitive conditions such as stroke, intercranial hemorrhage, or pneumothorax, for which early intervention is key. In the case of acute stroke due to large vessel occlusion, for example, studies have shown that an interventional radiology procedure called mechanical thrombectomy could achieve a favorable clinical outcome when performed within 4 to 6 hours of symptom onset [63]. This is where machine learning could be helpful. Leveraging the speed and the predictive power of machine learning, radiologists could use algorithmic outputs to prioritize cases that are deemed urgent.

True urgency, however, cannot be fully assessed until a case is opened and images are at least partially read. For this reason, many imaging clinics including those in the New York-Presbyterian hospital network tend to rely on the referring providers' communication of expectations as well as on their own insights, expertise and experience when prioritizing cases. In some sense, current practices rely on *human predictions* of urgency. The use of predictions powered by machine learning algorithms could augment current best practices and streamline the process of determining the order by which patient cases should be read.

But predictions, human-made or machine-learned, are rarely perfect. There will always exist never-before-seen cases that further compound the error. Good predictions have the potential to expedite the detection and treatment of time-sensitive conditions, but mispredictions could cause delays that are extremely costly. Given this understanding, the central question that we ask in this chapter is, *how can we take advantage of predictions to improve radiologists' workflow while accounting for prediction error?*

We abstract the setting described above and model it as a single-machine scheduling problem. A radiologist tasked with reviewing patient cases can be viewed as a single machine that is able to process one job at a time. Case urgencies are captured in the form of job weights, where the higher the weight, the greater the urgency. Patient treatment plans are often established upon completion of case review from the radiology department, so a natural objective would be to minimize the total sum of urgency-weighted completion times across all patients.

This single-machine problem of minimizing the weighted sum of job completion times is a decades-old problem that has already been extensively studied (see [78, 20], for example). In this chapter, we study this problem with the addition of a key feature: imperfect predictions of urgency. Patient cases randomly arrive into the system. At each job's time of arrival, we observe its predicted level of urgency given by some black-box predictive mechanism. True urgency is unknown and unobservable at this time, so priority decisions are necessarily made based on imperfect information. However, when radiologists are working through a patient case, interpreting the associated images and deciding if an abnormality is present, they are also gradually learning whether or not the case on hand is truly urgent (or non-urgent). Anecdotal evidence suggests that an image study is roughly a process of elimination via inspection from different angles [48, 76], so it is likely that a job's true urgency is known even before its processing is complete. We therefore allow radiologists to *preempt* a job midway then return to the remaining work for completion at a later point in time. With a preemptive strategy, we have an opportunity to hedge against prediction error by responding early to what is in hindsight a suboptimal decision made in the face of less-than-perfect information. We aim to find a policy for deciding which job or remainder thereof to process at any given time so that the total expected urgency-weighted sum of job completion times is minimized.

Our scheduling formulation allows for a wide range of models for describing the problem setting in ways that more accurately reflect clinical settings in the real world. For example, by using job weights to capture case urgencies, we are able to handle granularity in prioritization schemes beyond a binary classification of urgent vs. non-urgent. Our preemptive framework also allows flexibility in modeling the many different ways in which radiologists gain information as they process each patient case. Nevertheless, in this chapter, we focus on a highly stylized version of this model. We assume that each job can be categorized as one of two types: urgent or non-urgent. All jobs are available before any decisions are made, and we are able to observe each job's predicted priority class at this time. We further assume that every job shares the same processing time requirement. Without loss of generality, we assume unit processing time requirements. A fixed parameter $\alpha \in (0, 1)$ is used to denote the fraction of a job that must be processed before we learn its true type. We call this time point a job's α-point. In our model, we allow preemptions to occur only at these α-points, and assume that the residual work needed to complete an interrupted job is exactly the same as if the job had not been interrupted.

Our problem of scheduling with predictions is an exercise in online decision making even when all jobs are available to us in advance. Decisions are made with incomplete information in the form of imperfect predictions, to which we respond over time based on our observations of true job types. Classic results in online decision making have focused on finding solutions that are robust with provably good performance guarantees over all possible inputs and even in the worst case. An emerging line of research in this area leverages predictions to design algorithms that not only remain robust to worst-case inputs but also achieve performance guarantees that improve with prediction accuracy (see [62] for a survey). We continue this line of research and extend it to our problem setting. In what follows, we first find a threshold-based policy for deciding which job to process at any given time, and show that our proposed policy is the best possible over all non-anticipating policies. We then show that performance guarantees for this policy degrades gracefully as a function of prediction error. Our results indicate that our policy simultaneously achieves consistency (improvement with prediction accuracy) and robustness (protection against the worst case).

Related Work There has been an explosion of research activity in recent years that seek to augment online algorithms with machine-learned predictions. In this framework, the goal is to design algorithms with near-optimal performance when predictions are accurate while maintaining prediction-less guarantees in the worst case. The idea is that good predictions can help circumvent worst-case behavior. Classic optimization problems that are being reexamined under this framework include caching [57], matching [26, 5, 18, 51], secretary [28], knapsack [44] and facility location [4]. Problems in Nash social welfare [12], mechanism design [88] and revenue management [10] are also actively being studied in this context.

In online scheduling, problems that are being newly examined with learning augmentation include problems for minimizing average flow time [61, 62, 7], average completion time [69, 45], average weighted completion time [55], and makespan [50, 9]. Many of these studies with min-sum objectives assume that job processing requirements are not known to us in advance. In these settings, it is natural to use predictions of individual job processing times [69, 45, 61, 62, 7]. More recent work examines the use of permutation predictions, directly predicting algorithmic actions rather than input characteristics [55]. Our work studies the min-sum weighted completion time objective using predictions of an input characteristic that has not been considered in previous work: job weights.

Job weights are used to capture urgencies or priorities in our problem setting. Outside the realm of online scheduling with predictions, there is an extensive body of work that investigates the effect of priority classes. In the context of prediction error, our work is closely related to the works of Argon and Ziya [6] and McLay and Mayorga [58]. In a priority queue model, Argon and Ziya [6] make priority assignments for arriving customers based on imperfect indicators of priority types. The signal available to the decision maker is the probability that a customer is high priority. McLay and Mayorga [58] study the problem of dispatching ambulances when operators make classification errors in assessing patient risk via a Markov Decision Process. Our model is fundamentally different not just in framework, but more importantly in that we respond dynamically to real-time information gained while processing each job. Despite these major modeling differences, there are striking similarities in some of the insights and conclusions we draw. With Argon and Ziya [6], we share the same optimal policy structure given two priority classes with linear waiting costs. Both works have a signal (or, in our case, prediction)-based thresholding policy with strong ties to the generalized $c\mu$ rule. Our threshold policy also reveals how prediction quality impacts decision making. Similar insights are given by McLay and Mayorga [58] on when to over- or under-respond to perceived patient risk based on rates of classification error. It is clear that there are connections in our approaches despite their differences. In future work, it would be interesting to see when and how these frameworks converge.

In other related work, van der Zee and Theil [83] directly model misclassification rates in a single-server queue where priority assignments are made based on a probabilistic classifier. Steady-state results are derived when classification errors are known and very small. Singh et al. [77] eliminates the use of priority types as a middle-man altogether and directly prescribes placement into the priority queue. Finally, very recent work by Thompson et al. [80] explores the impact of prediction-driven prioritization schemes using a preemptive priority queue. Their simulated clinical impact assumes a fixed prediction error based on the expected diagnostic performance of machine-learned algorithms.

The remainder of this chapter is organized as follows. In Section 2.1, we introduce our model as well as a scheduling formulation of the problem. We present our main results in Section 2.2, where we show that a simple threshold-based policy is in fact the best possible in certain stylized settings. Section 2.3 extends this idea to a number of settings that more accurately reflect realistic scenarios. Finally, we conclude and lay out some additional thoughts for future research in Section 2.4.

2.1 Problem Formulation

We have a set of patient cases that must be processed by a radiologist. At time of arrival, each patient case is labeled with its predicted urgency level. These labels are observable. At every decision point, the radiologist decides which patient case to process. After processing a pre-specified fraction of a patient case, the radiologist learns the true priority of the case on hand and has the option to preempt that case in favor of another patient case. We capture this decision making process with a preemptive scheduling model. Our goal is to find a policy for minimizing the expected urgency-weighted sum of completion times across all patients. We describe the problem data and model, followed by a scheduling formulation of the problem.

Problem Data We have one radiologist (a single machine) processing patient cases (jobs) indexed by $[n] = \{1, \ldots, n\}$. A machine can only process one job at a time, and each job requires 1 unit in processing time. All jobs are assumed available at time 0 in advance of any decision making, i.e., release dates $r_j = 0$ for each job $j \in [n]$.

There are two priority classes, type 0 (urgent) and type 1 (non-urgent), each with its associated cost per unit delay (weights) ω_0 and ω_1, respectively, where $\omega_0 > \omega_1 > 0$. Each job is independently an urgent job with probability $\rho \in (0, 1)$, which we assume is known based on historical data. Job j's true urgency $true(j) \in \{0, 1\}$ is unknown a priori, and is revealed only after partially completing some fixed $\alpha \in (0, 1)$ fraction of the job. On the other hand, its predicted priority $pred(j) \in \{0, 1\}$ is immediately observable at its release date r_j. A binary classification system predicts the urgency level of each job independently according to the following probability matrix.

	predicted 0	predicted 1
true 0	$1 - \varepsilon_0$	ε_0
true 1	ε_1	$1 - \varepsilon_1$

Table 2.1: Prediction probability matrix

The probability of misclassifying a true type 0 job is the false negative rate ε_0, and the probability of misclassifying a true type 1 job is the false positive rate ε_1. We assume that $\varepsilon_0 \leq 1/2$ and $\varepsilon_1 \leq 1/2$, and that these prediction errors are known. We expect that they could be inferred from historical data or from expected generalization error rates associated with the machine learning algorithm that we use.

By Bayes' rule, job j is a type 0 job with probability p_j, where

$$p_j = \mathbb{P}(true(j) = 0 | pred(j) = 0) = \frac{(1-\varepsilon_0)\rho}{(1-\varepsilon_0)\rho + \varepsilon_1(1-\rho)} \tag{2.1}$$

if job j is predicted to be of high priority, and

$$p_j = \mathbb{P}(true(j) = 0 | pred(j) = 1) = \frac{\varepsilon_0\rho}{\varepsilon_0\rho + (1-\varepsilon_1)(1-\rho)} \tag{2.2}$$

otherwise. It is easy to verify that $\mathbb{P}(true(j) = 0|pred(j) = 0) \geq \mathbb{P}(true(j) = 0|pred(j) = 1)$ given our assumptions that ε_0 and ε_1 are both at most one half. Finally, the weight of job j is

$$\begin{aligned} w_j &= \omega_0 \cdot \mathbf{1}\left\{true(j) = 0\right\} + \omega_1 \cdot \mathbf{1}\left\{true(j) = 1\right\} \\ &= \omega_1 + (\omega_0 - \omega_1) \cdot \mathbf{1}\left\{true(j) = 0\right\}. \end{aligned} \tag{2.3}$$

Assumption 1. $\omega_1 < \omega_0(1-\alpha)$ *holds.*

Intuitively, Assumption 1 ensures that there is a large enough weight differential between urgent and non-urgent jobs to make preemption meaningful. The technical reasons for making this assumption will be discussed in the next section when it becomes relevant.

Model A decision point occurs whenever a job completes or a job's true priority is revealed. We call the latter decision point an α-point. Our model allows *preemptions*; at each α-point, we can either complete the job immediately, or preempt then process the remaining $1-\alpha$ units of work at a later point in time.

At each decision point t, we observe the state, which consists of the set of unopened jobs sorted in some order, and the set of partially processed jobs of which true types are already known. Of unopened jobs, only predicted priorities are known. Based on the state, we decide whether to open a new job of as-yet-unknown urgency or complete a job of known priority that only has $1-\alpha$ units of work remaining. Our decisions at each decision point are therefore made based on the *predicted* priorities of unopened jobs and the *true* priorities of partially processed jobs. If we decide to open a new job, we process a job chosen according to some predetermined order and meet our next decision point at the next α-point $t+\alpha$, at which time we observe the job's true type. We then update the state by moving this job from the set of unopened to the set of partially processed. Otherwise, we complete a job at $t+(1-\alpha)$, incur a weighted cost to the objective based on the true urgency of the job just completed, and remove that job from the system entirely.

Objective Each of the n arriving jobs is independently a high priority job with probability ρ, and is assigned a predictive label according to the probability matrix given in Table 2.1. Letting C_j denote the completion time of job j, our goal is to minimize $\mathbb{E}\left(\sum_{j=1}^{n} w_j C_j\right)$.

Scheduling Formulation We first consider the offline version of this problem in which jobs' true types are known a priori. This is a single-machine problem of minimizing the weighted sum of completion times, written $1||\sum w_j C_j$ in the scheduling notation of Graham et al. [40], and can be solved using the following result given by Smith [78].

Theorem 1 (Smith's WSPT Rule)**.** *For the single-machine problem of minimizing the* *weighted sum of completion times, the Weighted Shortest Processing Time (WSPT) rule is optimal.*

The WSPT rule sorts jobs in nonincreasing order of weight-to-processing-time ratios. Given our unit processing time assumption, sorting jobs in WSPT order is equivalent to sorting jobs in nonincreasing order of true priorities. Then, processing job j for completion at time j yields an optimal schedule, so $\mathbb{E}(\mathsf{OPT}) = \mathbb{E}\left(\sum_{j=1}^{n} jw_j\right)$.

Our problem, however, is a *non-clairvoyant* online decision making problem in which jobs' true priorities are not known until jobs are at least partially processed. We follow the WSPT rule and sort jobs in nonincreasing order of *predicted* weights, breaking ties arbitrarily. We then proceed by opening jobs in this sorted order. The rest of this chapter is focused on showing that the performance gap between the online and offline versions of this problem can be reduced with the use of *predictions*, especially when the predictor has low error.

A Toy Example Consider the following deterministic 9-job example where each column represents a single job.

true types (unknown)	0	0	0	0	1	1	1	1	1
predicted types (observed)	0	1	0	0	0	1	1	1	0

At each job's release date, we observe its predicted type. True types are not known at this time. We proceed by sorting jobs in WSPT order of predicted priorities.

In this example, there are $\binom{5}{2}\binom{4}{1}$ possible permutations by which we could open jobs as a result of this WSPT sort, driven by mispredictions. One such

predicted types (sorted)	0	0	0	0	0	1	1	1	1
true types (permutation π)	0	1	0	0	1	1	1	0	1

Table 2.2: An example of a possible job ordering

permutation π is given as an example above. Once a job is opened, we learn its true type after processing α units of the job. At this α-point, we have the option of either completing the remaining $1 - \alpha$ units of work, or opening the next job in π. Decisions are made over time with the goal of minimizing $\mathbb{E}\left(\sum_{j=1}^{n} w_j C_j\right)$ across all possible permutations of job orderings.

2.2 The β-Threshold Rule: An Optimal Policy

We provide an optimal policy for our problem in this section. Before we do so, we first discuss an old scheduling result by Schrage [74].

Theorem 2 (Schrage's SRPT Rule)**.** *In the preemptive single-machine problem of minimizing the sum of completion times, where jobs are arriving over time, the Shortest Remaining Processing Time (SRPT) rule is optimal.*

By the SRPT rule, given any two jobs of the same weight, we should process the job with the shorter amount of remaining work first. SRPT applied to our problem confirms a general intuition that it is never optimal to preempt a job that is revealed to be of high priority; type 0 jobs will always be processed non-preemptively. This does not change our model, but it does help simplify some aspects of it. Since preemption only occurs on type 1 jobs, we are able to eliminate α-points with respect to type 0 jobs. It also suffices to track the number of partially processed jobs as these jobs are all of type 1.

We now present a thresholding policy that minimizes our objective across all non-anticipating policies. Without loss of generality, we sort jobs according to their predicted priorities, breaking ties arbitrarily. This is equivalent to sorting jobs in nonincreasing order of their type 0 probabilities p_j as defined in equations (2.1)-(2.2). As we proceed with our policy, we open jobs in this order. Define a constant

$$\beta = \frac{\alpha}{1-\alpha} \cdot \frac{\omega_1}{\omega_0 - \omega_1}.$$

At each decision point, we observe the state $(\mathcal{S}, \ell)$, where $\mathcal{S}$ is the set of unopened jobs and ℓ is the number of partially processed type 1 jobs. If either $\mathcal{S} = \emptyset$ or $\ell = 0$, we do not have a decision to make; if the former, we complete a partially processed type 1 job, and if the latter, we open a new job. If $\mathcal{S} = \emptyset$ and $\ell = 0$, we are done. Thus, we assume that $\ell > 0$ and $\mathcal{S} \neq \emptyset$ with $k = \min(\mathcal{S})$, which means that job k is the next job in line. We make our decisions by comparing p_k against β: if $p_k \leq \beta$, we process the remaining $1-\alpha$ units of a partially completed low priority job and reach our next decision point at completion. If $p_k > \beta$, we open job k and process α units of the job, at which time we learn of job k's true type. If job k is a type 1 job, we are at our new decision point. Otherwise, we process job k to completion for another $1-\alpha$ units and make our next decision when job k's processing is complete.

Theorem 3. *The β-threshold rule is optimal.*

Proof. Suppose on the contrary that there exists an optimal policy that does not follow the β-threshold rule. By assumption, if we run this policy on any instance of our problem input, there exists at least one decision point where the optimal policy observes the given state and makes a decision that deviates from ours. We consider the *last* such decision point and call it time t, and further assume

Algorithm 1 The β-Threshold Rule

```
Require: jobs sorted in nonincreasing order of $p_j$
Initialize:
    $t \leftarrow 0$                                   ▷ time
    $\mathcal{S} \leftarrow [n]$                       ▷ set of unopened jobs
    $\ell \leftarrow 0$                                ▷ number of partially completed type 1 jobs
procedure CompleteLow($t, \mathcal{S}, \ell$)          ▷ complete a known low priority job
    $\ell \leftarrow \ell - 1$
    $t \leftarrow t + (1 - \alpha)$
end procedure
procedure OpenNext($t, \mathcal{S}, \ell$)             ▷ open a new job, then stop at the $\alpha$-point
    $k \leftarrow \min(\mathcal{S})$
    $\mathcal{S} \leftarrow \mathcal{S} \setminus \{k\}$
    $t \leftarrow t + \alpha$
    if $true(k) = 0$ then                              ▷ nonpreemptively complete a type 0 job
        $t \leftarrow t + (1 - \alpha)$
    else
        $\ell \leftarrow \ell + 1$
    end if
end procedure
while $(\mathcal{S}, \ell)$ is not $(\emptyset, 0)$ do
    if $\mathcal{S} = \emptyset$ then
        CompleteLow($t, \mathcal{S}, \ell$)
    else if $\ell = 0$ then
        OpenNext($t, \mathcal{S}, \ell$)
    else
        $k \leftarrow \min(\mathcal{S})$
        if $p_k > \beta$ then
            OpenNext($t, \mathcal{S}, \ell$)
        else
            CompleteLow($t, \mathcal{S}, \ell$)
        end if
    end if
end while
```

that the observed state at that time is $(\mathcal{S}, \ell)$, where $\mathcal{S}$ is the set of unopened jobs such that $k = \min(\mathcal{S})$ and $\ell > 0$ is the number of partially processed type 1 jobs.

Two things may have occurred at time t: $p_k > \beta$ and the optimal policy processes a low priority job for completion at time $t + (1 - \alpha)$, or $p_k \leq \beta$ and the optimal policy proceeds by opening job k. In both cases, we show that choosing the alternative improves the objective value and ensures that the resulting schedule is consistent with the β-threshold rule.

i $p_k > \beta$: according to the β-threshold rule, we should have opened job k at time t; the optimal policy decided otherwise and completed a type 1 job (let us call this job i) at time $t + (1 - \alpha)$. We proceed by identifying another point in time in the schedule generated by the optimal policy to process job i, which would allow job k to be processed at time t instead. We then show by an interchange argument that doing so improves the objective value.

Starting from time t, trace time forward in the schedule generated by the optimal policy. Since t is the last decision point that deviates from the β-threshold rule by assumption and the set of unopened jobs at the next decision point $t + (1 - \alpha)$ remains unchanged so that $k = \min(\mathcal{S})$, the optimal policy opens job k at time $t+(1-\alpha)$. We continue to trace time forward until some time u, when the optimal policy begins processing the remaining $1-\alpha$ units of a previously preempted, true low priority job for the first time since completing job i at time $t + (1 - \alpha)$. We show that within the interval $[t, u)$, we can improve the objective by delaying the completion of job i to $C_i = u$ and moving up the schedule in $[t + 1 - \alpha, u)$ by $1 - \alpha$ units to $[t, u - 1 + \alpha)$.

By our assumptions, u is the time at which either $\mathcal{S} = \emptyset$ (every job has been opened), or the next job's type 0 probability falls below β, whichever happens first. Therefore, all jobs opened in $[t+1-\alpha, u)$ are above the probability threshold β. Let $z \geq 1$ denote the number of jobs opened in $[t+1-\alpha, u)$ including job k. Among these z jobs, suppose there are z_0 jobs of true type 0 that are completed immediately where $z_0 = \sum_{j=k}^{k+z-1} \mathbf{1}\{true(j) = 0\}$. The remaining $z - z_0$ jobs are revealed to be of true type 1 after α units of processing, then are preempted. These preempted jobs are not processed until at least time u.

By interchange, each of the z_0 type 0 jobs complete $1-\alpha$ units earlier. On the other hand, completion of job i, a type 1 job, is delayed by $u - (t+1-\alpha) = z_0 + (z - z_0)\alpha$, which sums one unit of delay for every completed type 0 job and an α unit of delay for every preempted type 1 job. None of the other jobs are affected by this interchange. Thus, the overall change to the objective is

$$\begin{aligned} & -z_0\omega_0(1-\alpha) + \omega_1\left(z\alpha + z_0(1-\alpha)\right) \qquad (2.4) \\ = \quad & -z_0(\omega_0 - \omega_1)(1-\alpha) + \alpha\omega_1 z \\ = \quad & -(\omega_0 - \omega_1)(1-\alpha)\left(z_0 - \frac{\alpha}{1-\alpha} \cdot \frac{\omega_1}{\omega_0 - \omega_1} z\right) \\ = \quad & -(\omega_0 - \omega_1)(1-\alpha)\left(z_0 - \beta z\right) \\ = \quad & -(\omega_0 - \omega_1)(1-\alpha) \sum_{j=k}^{k+z-1} \left(\mathbf{1}\{true(j) = 0\} - \beta\right). \end{aligned}$$

In expectation,

$$-(\omega_0 - \omega_1)(1-\alpha) \sum_{j=k}^{k+z-1} (p_j - \beta) < 0$$

since $p_j > \beta$ for each $j = k, \ldots, k+z-1$, which establishes a contradiction. We have also shown how to choose the interval $[t, u)$ for interchange to ensure that the schedule is consistent with the β-threshold rule from time t onward.

ii $p_k \leq \beta$: according to the β-threshold rule, we should have completed a type 1 job for completion at time $t+1-\alpha$. Instead, the optimal policy opened job k. The proof proceeds similarly to the above in that we first identify an appropriate point in time in the schedule generated by the optimal policy to open job k, then use interchange. The main difference lies in that the amount of delay from opening job k is not immediately clear, since that depends on job k's true type.

We first argue that in fact, regardless of type, job k completes at $t+1$ in the schedule generated by the optimal policy. This is trivially true if job k is a type 0 job. Otherwise, the optimal policy meets its next decision point at $t+\alpha$, where the set of unopened jobs is $\mathcal{S} = [n] \setminus [k]$ and there are now $\ell+1$ true type 1 jobs that are not yet fully processed including job k. By our assumption that time t is the last decision point at which the optimal policy deviates from the β-threshold rule, the optimal policy completes a type 1 job at $(t+\alpha)+(1-\alpha) = t+1$ since $\beta \geq p_k \geq p_{k+1}$ at time $t+\alpha$. We are free to label this job as job k. Then, starting from $t+1$, the optimal policy will complete the remaining ℓ type 1 jobs in succession, completing the last type 1 job at time $t+\ell(1-\alpha)+1$.

We show that within the interval $[t, t+\ell(1-\alpha)+1)$, we can improve the objective by delaying the opening of job k to time $t+\ell(1-\alpha)$, when ℓ type 1 jobs have each completed the remaining $1-\alpha$ units of work. Even with this interchange, since $\beta \geq p_{k+1}$, job k will be processed nonpreemptively regardless of type so that $C_k = t+\ell(1-\alpha)+1$. None of the other jobs are affected. The overall change to the objective is

$$\begin{aligned}
& -\ell\omega_1 + w_k\ell(1-\alpha) && (2.5)\\
= \quad & -\ell\omega_1 + (\omega_1 + (\omega_0-\omega_1)\cdot \mathbf{1}\left\{true(k)=0\right\})\,\ell(1-\alpha) && \text{by (2.3)}\\
= \quad & -\ell\omega_1 + \omega_1\ell(1-\alpha) + (\omega_0-\omega_1)\mathbf{1}\left\{true(k)=0\right\}\ell(1-\alpha) &&\\
= \quad & -\ell\alpha\omega_1 + (\omega_0-\omega_1)\mathbf{1}\left\{true(k)=0\right\}\ell(1-\alpha) &&\\
= \quad & -\ell(\omega_0-\omega_1)(1-\alpha)\left(\frac{\alpha}{1-\alpha}\cdot\frac{\omega_1}{\omega_0-\omega_1} - \mathbf{1}\left\{true(k)=0\right\}\right) &&\\
= \quad & -\ell(\omega_0-\omega_1)(1-\alpha)\left(\beta - \mathbf{1}\left\{true(k)=0\right\}\right). &&
\end{aligned}$$

In expectation,

$$-\ell(\omega_0-\omega_1)(1-\alpha)\left(\beta - p_k\right) \le 0$$

since $p_k \le \beta$ by assumption. The resulting schedule is consistent with the β-threshold rule from time t onward. This establishes the desired contradiction and concludes the proof. □

Given our results in Theorem 3, we now provide a technical reason behind Assumption 1 which requires $\omega_1 < \omega_0(1-\alpha)$. Suppose on the contrary that $\omega_1 \ge \omega_0(1-\alpha)$. Rearranging inequalities, this also implies that $\alpha \ge 1-\omega_1/\omega_0$. Then,

$$\beta = \frac{\alpha}{1-\alpha}\cdot\frac{\omega_1}{\omega_0-\omega_1} \ge \frac{\alpha\omega_0}{\omega_0-\omega_1} = \frac{\alpha}{1-\omega_1/\omega_0} \ge 1$$

and so by the β-threshold rule we would complete every job nonpreemptively. Because this is not a particularly interesting case, we focus our efforts where preemption offers room for improvement. All the same, we provide a performance upper bound for this case in Corollary 5.

The β-threshold rule may seem arbitrary at first, but there is an intuitive explanation for it that reveals a strong connection with the celebrated $c\mu$ rule. Recall that for any job j, $\mathbb{E}(w_j) = \omega_1 + (\omega_0-\omega_1)p_j$ by Equation (2.3).

Proposition 1.

$$\frac{\mathbb{E}(w_j)}{1} > \frac{\omega_1}{1-\alpha} \iff p_j > \beta.$$

Proof. Expanding the left hand side of the inequality,

$$\begin{aligned}(1-\alpha)\left(\omega_1 + (\omega_0 - \omega_1)p_j\right) > \omega_1 &\iff (1-\alpha)(\omega_0 - \omega_1)p_j > \alpha\omega_1 \\ &\iff p_j > \frac{\alpha}{1-\alpha} \cdot \frac{\omega_1}{\omega_0 - \omega_1} \\ &\iff p_j > \beta,\end{aligned}$$

which is equivalent to the conditions given in the β-threshold rule. Inequality in the other direction holds analogously. □

At every decision point, applying the β-threshold rule is equivalent to comparing the $c\mu$ of an unopened job j against the $c\mu$ of a known low priority job with $1-\alpha$ units of residual work, and choosing the job with the higher $c\mu$ value.

Depending on our chosen parameter values, the β-threshold rule may give rise to three modes of decision making: a *nonpreemptive* policy, a *preemptive* policy, and a *hybrid* policy that switches from a preemptive policy to a nonpreemptive policy sometime in between.

Let us first assume that jobs are sorted in WSPT order of predicted priorities. A *nonpreemptive* policy completes every job in sorted order without preemption. A schedule generated by a nonpreemptive policy is a nonpreemptive schedule. A policy is *preemptive* if, opening jobs in sorted order, every type 1 job is preempted at its α-point. These low priority jobs will only be revisited once all n jobs have been opened and every high priority job has completed its processing. The resulting schedule is a preemptive schedule. Preemptive and nonpreemptive schedules are two non-adaptive special cases of a schedule generated by the β-threshold rule.

The *hybrid* policy, on the other hand, is an adaptive policy that switches between the preemptive and nonpreemptive regimes based on the predictive label of the job being processed. More specifically, a preemptive strategy is used on jobs that are expected to be type 0, while a nonpreemptive strategy is used on the remaining jobs that are predicted to be non-urgent. Given our initial sort, we make this switch exactly once.

In what follows, we specify the conditions that give rise to each of our policies.

Corollary 1. *A nonpreemptive policy is optimal if*

$$\min\left(\rho(1-\beta), \beta(1-\rho)\right) \leq \rho(1-\beta)\varepsilon_0 + \beta(1-\rho)\varepsilon_1 \textit{ and } \rho \leq \beta.$$

Proof. The statement follows directly from Theorem 3. We employ a nonpreemptive policy if $\beta \geq \mathbb{P}(true(\cdot) = 0|pred(\cdot) = 0)$ where the probability is as defined in (2.1)-(2.2). Rearranging the inequality, we obtain the result. □

Corollary 2. *A preemptive policy is optimal if*

$$\min\left(\rho(1-\beta), \beta(1-\rho)\right) \leq \rho(1-\beta)\varepsilon_0 + \beta(1-\rho)\varepsilon_1 \textit{ and } \rho > \beta.$$

Corollary 3. *A hybrid policy is optimal if*

$$\rho(1-\beta)\varepsilon_0 + \beta(1-\rho)\varepsilon_1 < \min\left(\rho(1-\beta), \beta(1-\rho)\right).$$

The β-threshold rule admits a hybrid policy if

$$\mathbb{P}(true(\cdot) = 0|pred(\cdot) = 1) \leq \beta < \mathbb{P}(true(\cdot) = 0|pred(\cdot) = 0). \tag{2.6}$$

It follows naturally from Bayes' rule that the gap between the two conditional probabilities in (2.6) is large when prediction error is low. When that is the case, β is much more likely to fall in between these two probabilities for our chosen parameter values, resulting in an adaptive hybrid policy. On the other hand, when we have a predictor with high prediction error, this conditional probability gap is likely to be smaller, in which case a non-adaptive policy would be best.

2.2.1 Performance Analysis

We now quantify the performance of our policies as a function of prediction error. More specifically, we fix the number of urgent jobs among our n available jobs, then obtain exact expressions for expected performance conditional on this quantity, which we denote n_0. Performance is measured against the offline optimum OPT given n_0. This focus on conditional expectation allows us to remove one layer of randomness from our problem and isolate the effects of misprediction. The expressions we derive in this section will also be useful for competitive analysis in our next section. Extending our results to obtain expressions for unconditional expectations of performance can be easily done by using the first and second moments of n_0.

Proposition 2. *Let C_j^{ϕ} and C_j^{α} each denote job j's completion time in nonpreemptive* *and preemptive schedules, respectively. Given n_0,*

$$\mathsf{OPT} = (\omega_0 - \omega_1)\frac{n_0(n_0+1)}{2} + \omega_1\frac{n(n+1)}{2} \tag{2.7}$$

$$\mathbb{E}\left(\sum_{j=1}^{n} w_j C_j^{\phi} \middle| n_0\right) = \mathsf{OPT} + (\omega_0 - \omega_1)\mathbb{E}(X|n_0) \tag{2.8}$$

$$\mathbb{E}\left(\sum_{j=1}^{n} w_j C_j^{\alpha} \middle| n_0\right) = \mathsf{OPT} + \alpha\omega_0\mathbb{E}(X|n_0) + \alpha\omega_1\mathbb{E}(Y|n_0) \tag{2.9}$$

where, letting $n_1 = n - n_0$,

$$\mathbb{E}(X|n_0) = \frac{(\varepsilon_0+\varepsilon_1)n_0n_1}{2},\ \mathbb{E}(Y|n_0) = \frac{n_1(n_1-1)}{2}. \tag{2.10}$$

Proof. The offline optimum is easy to compute by WSPT:

$$\begin{aligned}\mathsf{OPT} = \sum_{j=1}^{n} jw_j &= \sum_{j=1}^{n_0}\omega_0 j + \sum_{j=n_0+1}^{n}\omega_1 j \\ &= (\omega_0 - \omega_1)\frac{n_0(n_0+1)}{2} + \omega_1\frac{n(n+1)}{2}.\end{aligned}$$

For (2.8) and (2.9), recall that sorting jobs in WSPT order of predicted priorities results in a number of possible permutations of true priorities. We evaluate the objective for some fixed permutation π of true types, then take the expectation across all possible permutations.

In a nonpreemptive schedule, each pair of jobs whose true types are out of order, i.e., a pair of (true 1, true 0), adds $\omega_0 - \omega_1$ to the objective relative to the offline optimum. Letting X denote the number of such inversions in π, $\sum_{j=1}^{n} w_j C_j^{\phi} = \mathsf{OPT} + (\omega_0 - \omega_1)X$.

In a preemptive schedule, the cost of one inversion is $\alpha\omega_0$, since a type 1 job preempts after processing α units and allows a type 0 job to be processed and completed before resuming its $1-\alpha$ units of residual work. This schedule also incurs a cost of $\alpha\omega_1$ for every pair of (true 1, true 1) jobs, because the policy requires that we open and preempt both jobs before we begin processing any remaining work for completion. Thus, letting Y denote the number of (true 1, true 1) pairs in π, $\sum_{j=1}^{n} w_j C_j^{\alpha} = \mathsf{OPT} + \alpha\omega_0 X + \alpha\omega_1 Y$.

We conclude the proof by computing $\mathbb{E}(X|n_0)$.

$$\begin{aligned}
X &= \sum_{j=1}^{n}\sum_{k>j} \mathbb{1}\left\{\pi(j)=1\right\}\mathbb{1}\left\{\pi(k)=0\right\} \\
&= \sum_{j=1}^{n}\sum_{k>j} \mathbb{1}\left\{\pi(j)=1, pred(j)=0\right\}\mathbb{1}\left\{\pi(k)=0, pred(k)=0\right\} \\
&\quad + \sum_{j=1}^{n}\sum_{k>j} \mathbb{1}\left\{\pi(j)=1, pred(j)=0\right\}\mathbb{1}\left\{\pi(k)=0, pred(k)=1\right\} \\
&\quad + \sum_{j=1}^{n}\sum_{k>j} \cancel{\mathbb{1}\left\{\pi(j)=1, pred(j)=1\right\}\mathbb{1}\left\{\pi(k)=0, pred(k)=0\right\}} \\
&\quad + \sum_{j=1}^{n}\sum_{k>j} \mathbb{1}\left\{\pi(j)=1, pred(j)=1\right\}\mathbb{1}\left\{\pi(k)=0, pred(k)=1\right\}
\end{aligned}$$

where the third term cancels because of our initial sort in WSPT order of predicted priorities. Accounting for the order of jobs, we can replace $\pi(\cdot)$ with $true(\cdot)$:

$$\begin{aligned}
X = \frac{1}{2}&\left(\sum_{j=1}^{n}\mathbb{1}\left\{true(j)=1, pred(j)=0\right\}\right)\left(\sum_{k=1}^{n}\mathbb{1}\left\{true(k)=0, pred(k)=0\right\}\right) \\
&+\left(\sum_{j=1}^{n}\mathbb{1}\left\{true(j)=1, pred(j)=0\right\}\right)\left(\sum_{k=1}^{n}\mathbb{1}\left\{true(k)=0, pred(k)=1\right\}\right) \\
&+\frac{1}{2}\left(\sum_{j=1}^{n}\mathbb{1}\left\{true(j)=1, pred(j)=1\right\}\right)\left(\sum_{k=1}^{n}\mathbb{1}\left\{true(k)=0, pred(k)=1\right\}\right).
\end{aligned}$$

Order in the second term is, again, automatically satisfied by how we sort the jobs. Taking the conditional expectation given n_0 and letting $n_1 = n - n_0$, we obtain

$$\begin{aligned}\mathbb{E}(X|n_0) &= \frac{\varepsilon_1(1-\varepsilon_0)n_0n_1}{2} + \varepsilon_0\varepsilon_1 n_0 n_1 + \frac{\varepsilon_0(1-\varepsilon_1)n_0n_1}{2} \\ &= \frac{(\varepsilon_0+\varepsilon_1)n_0n_1}{2}.\end{aligned} \tag{2.11}$$

Finally, $\mathbb{E}(Y|n_0) = \binom{n_1}{2} = n_1(n_1-1)/2$. □

Given the proposition above, we can combine (2.8)-(2.9) to give expressions for the performance of the β-threshold rule. In essence, the β-threshold rule dictates when to move from a preemptive regime to a nonpreemptive regime. Based on our previous analyses, this cutoff occurs once we complete the last job that is predicted to be of high priority.

Proposition 3. *Let C_j^β denote job j's completion time in a schedule generated by the β-threshold rule. This schedule is nonpreemptive with performance given in (2.8) if $\beta \geq \max_j p_j$, and preemptive with performance given in (2.9) if $\beta < \min_j p_j$. Otherwise, relative to* OPT *as defined in (2.7), the conditional expectation given n_0 is*

$$\mathbb{E}\left(\sum_{j=1}^{n} w_j C_j^\beta \middle| n_0\right) = \mathsf{OPT} + \mathbb{E}\left(\underbrace{\alpha\omega_0 X_0 + \alpha\omega_1 Y_0}_{\textit{preemptive}} + \underbrace{(\omega_0-\omega_1)(X-X_0)}_{\textit{nonpreemptive}} \middle| n_0\right) \tag{2.12}$$

where, letting $n_1 = n - n_0$,

$$\mathbb{E}(X_0|n_0) = \frac{\varepsilon_1(1-\varepsilon_0)n_0n_1}{2},\ \mathbb{E}(Y_0|n_0) = \frac{\varepsilon_1^2\,(n_1^2-n_1)}{2}$$

and $\mathbb{E}(X|n_0)$ is as defined in (2.10).

Proof. X_0 and Y_0 count the number of (true 1, true 0) and (true 1, true 1) pairs, respectively, from the set of jobs that are predicted to be of type 0. The expected value of X_0 given n_0 is given in the first term of (2.11) in Proposition 2. The expected value of Y_0 given n_0 is the expected value of $\binom{\mathsf{Bin}(n_1,\varepsilon_1)}{2}$ where $\mathsf{Bin}(n_1, \varepsilon_1)$ denotes a binomial random variable with parameters n_1 and ε_1. The remainder of the proof is identical to the one given in the proposition above. □

The expression in (2.12) makes it clear that the impacts of false positive and false negative rates to performance may vary.

Corollary 4. *If the false positive rate $\varepsilon_1 = 0$, a hybrid policy gives a nonpreemptive schedule.*

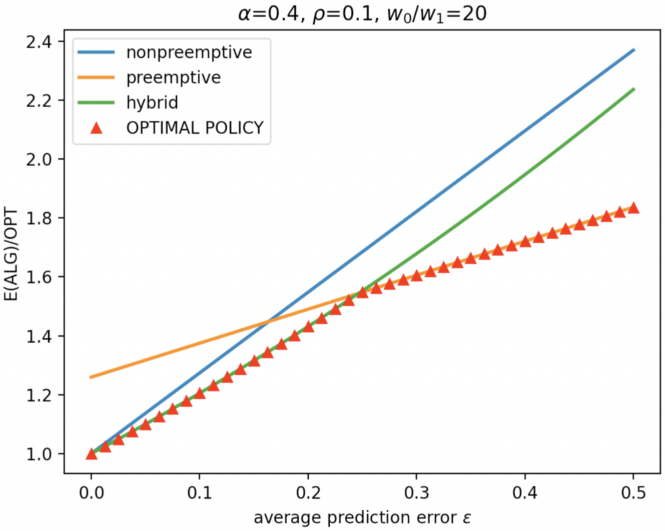


Figure 2.1: Expected performance

Figure 2.1 plots the unconditional expected performance of each of our policies as a function of prediction error, where performance is normalized by the offline optimum. For illustrative purposes, we assume $\varepsilon_0 = \varepsilon_1$ and choose parameter values of $\alpha = 0.4$, $\rho = 0.1$, and $\omega_0/\omega_1 = 20$. Since our problem is a minimization problem, the lower the ratio of $\mathbb{E}(\mathsf{ALG})/\mathsf{OPT}$, the better.

The nonpreemptive policy performs very well when prediction error is low, in fact recovering the offline optimum when we are given perfect predictions.

This policy blindly trusts the predictor, however, resulting in poor performance when prediction quality is low. On the other hand, the preemptive policy opts *not* to trust the predictions and searches for high priority jobs regardless of the advice it receives. It performs well when prediction quality is low, but is overly aggressive against non-urgent jobs when predictions are accurate, penalizing them unnecessarily. The β-threshold rule takes the best of both worlds. When prediction error is low, our optimal policy strategically shifts from a preemptive to a nonpreemptive policy, outperforming each of the individual non-adaptive policies. Once prediction error reaches a certain point and predictive labels lose meaning, the β-threshold rule shifts to a preemptive policy.

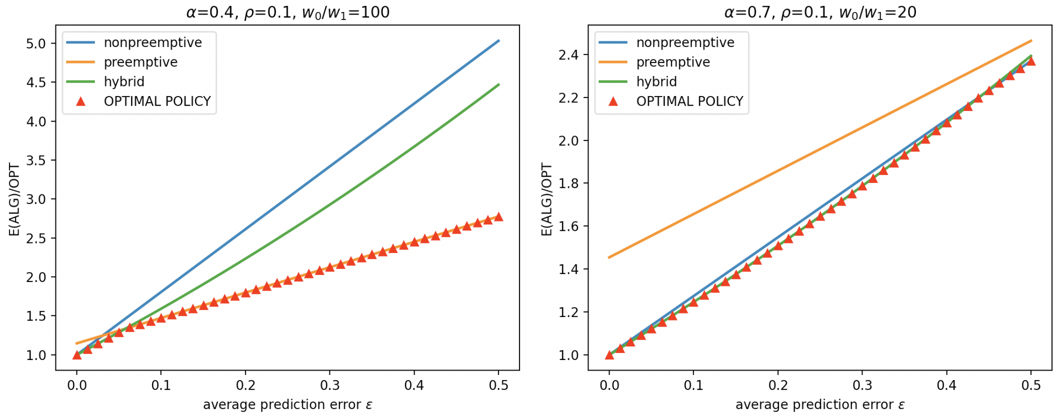


Figure 2.2: Expected performance

Needless to say, performance depends heavily on our chosen parameter values. Figure 2.2 gives two examples in which performance improvements from the β-threshold rule are modest at best. The plot on the left panel considers a case where relative priority values are set very high at $\omega_0/\omega_1 = 100$. Analytically, our chosen parameter values push down the β value significantly so that it becomes unlikely that β will fall between the two conditional probabilities given in (2.6) unless the predictor is very accurate. Intuitively, the relative priority of urgent jobs is so great that there is simply no room for prediction error. This explains the low tolerance for error before our optimal policy switches from a hybrid policy to a preemptive policy. The hybrid policy still outperforms both non-adaptive policies when predictions are accurate.

The second plot in Figure 2.2 is a rare example in which a nonpreemptive policy outperforms a preemptive policy throughout. Here, we consider a high value of α where $\alpha = 0.7$, which pushes up the β value and the cost of preemption at the same time. In this case, the high cost of preemption makes it preferable to complete a low priority job than to open a new job that may or may not be high priority. We again observe that the hybrid policy outperforms both non-adaptive policies, but the improvements are small.

2.2.2 Competitive Analysis

We provide performance guarantees for our policies in this section.

Definition 1. *The* competitive ratio *of an online algorithm* ALG *is* CR *if the inequality*

$$\mathbb{E}(\mathsf{ALG}) \leq \mathsf{CR} \cdot \mathsf{OPT}$$

holds for all possible inputs. Then, we can also say that ALG *is* CR-competitive*.*

In our competitive analysis, an adversary deliberately choosing a difficult input has control over the mix of urgent and non-urgent jobs. Let q denote the fraction of urgent jobs among all available jobs. We shall aim to find the worst case values of q. This analysis addresses a known weakness in our model. Our model assumes that each arriving job is independently a high priority job with probability $\rho \in (0, 1)$, and our proposed β-threshold rule is optimal with respect to this parameter. While ρ could be inferred from historical data, it also tends to be highly volatile and sensitive to environmental changes. Mass casualty events or insurance policy changes are some examples that could drive the value of ρ up or down. With competitive analysis, we are able to guarantee performance for all possible values of ρ. Furthermore, as a byproduct of our analyses, we can observe which values of ρ result in the worst case.

Lemma 1. *The performance of a nonpreemptive policy is bounded above by* $\mathsf{CR}^{\phi} \cdot \mathsf{OPT}$ *where*

$$\mathsf{CR}^{\phi} = 1 + \varepsilon \left(\sqrt{\frac{\omega_0}{\omega_1}} - 1 \right)$$

and $\varepsilon = (\varepsilon_0 + \varepsilon_1)/2$ *is the average of the false negative and false positive rates* ε_0 *and* ε_1.

Proof. Let C_j^{ϕ} denote job j's completion time in a nonpreemptive schedule. Based on (2.8),

$$\frac{\mathbb{E}\left(\sum_{j=1}^{n} w_j C_j^{\phi} \middle| n_0 \right)}{\mathsf{OPT}} - 1 = \frac{(\omega_0 - \omega_1)\mathbb{E}(X|n_0)}{\mathsf{OPT}}.$$

Expanding the terms as given in Proposition 2 and letting $q = n_0/n$,

$$\begin{aligned} \frac{(\omega_0 - \omega_1)\mathbb{E}(X|n_0)}{\mathsf{OPT}} &= \frac{2\varepsilon(\omega_0 - \omega_1)n_0(n - n_0)}{(\omega_0 - \omega_1)n_0(n_0 + 1) + \omega_1 n(n + 1)} \\ &= \frac{2\varepsilon(\omega_0 - \omega_1)q(1 - q)n^2}{(\omega_0 - \omega_1)(q^2n^2 + qn) + \omega_1(n^2 + n)} \end{aligned}$$

then, we approach the limit from below as $n \to \infty$ so the upper bound is

$$\frac{2\varepsilon(\omega_0 - \omega_1)q(1 - q)}{(\omega_0 - \omega_1)q^2 + \omega_1}.$$

It is straightforward calculus to show that

$$\max_{0 \leq q \leq 1} \frac{2\varepsilon(\omega_0 - \omega_1)q(1 - q)}{(\omega_0 - \omega_1)q^2 + \omega_1} = \varepsilon \left(\sqrt{\frac{\omega_0}{\omega_1}} - 1 \right),$$

which implies the result. The maximum is attained by choosing

$$q = \sqrt{\frac{\omega_1}{\omega_0 - \omega_1} + \left(\frac{\omega_1}{\omega_0 - \omega_1}\right)^2} - \frac{\omega_1}{\omega_0 - \omega_1}.$$

□

An immediate consequence of the above lemma is a performance bound for the schedule when Assumption 1 does not hold. Recall that without Assumption 1, a nonpreemptive schedule is an optimal schedule.

Corollary 5. *If $\omega_1 \geq \omega_0(1-\alpha)$, the competitive ratio is $1 + \varepsilon\left(\sqrt{\frac{1}{1-\alpha}} - 1\right)$.*

Lemma 2. *The competitive ratio CR^α of a preemptive policy is*

$$\mathsf{CR}^\alpha = \begin{cases} 1+\alpha & \text{if } \varepsilon \leq \omega_1/\omega_0, \text{ and} \\ 1 + \dfrac{\alpha}{2}\dfrac{\omega_0}{\omega_0 - \omega_1}\left(1 - 2\varepsilon + \sqrt{1 - 4\varepsilon + 4\varepsilon^2\left(\dfrac{\omega_0}{\omega_1}\right)}\right) & \text{otherwise,} \end{cases}$$

where $\varepsilon = (\varepsilon_0 + \varepsilon_1)/2$ is the average of the false negative and false positive rates ε_0 and ε_1.

Proof. The first part of the proof proceeds similarly. Let C_j^α denote job j's completion time in a preemptive schedule. Letting $n_1 = n - n_0$ and $q = n_0/n$,

$$\begin{aligned} \frac{\mathbb{E}\left(\sum_{j=1}^n w_j C_j^\alpha \middle| n_0\right)}{\mathsf{OPT}} - 1 &= \frac{\alpha\omega_0\mathbb{E}(X|n_0) + \alpha\omega_1\mathbb{E}(Y|n_0)}{\mathsf{OPT}} \\ &= \alpha \cdot \frac{2\varepsilon\omega_0 n_0 n_1 + \omega_1 n_1(n_1 - 1)}{(\omega_0 - \omega_1)n_0(n_0+1) + \omega_1 n(n+1)} \\ &= \alpha \cdot \frac{2\varepsilon\omega_0 q(1-q)n^2 + \omega_1\left((1-q)^2 n^2 - (1-q)n\right)}{(\omega_0 - \omega_1)\left(q^2n^2 + qn\right) + \omega_1(n^2+n)} \end{aligned}$$

then, we approach the limit from below as $n \to \infty$ so the upper bound is

$$\alpha \cdot \frac{2\varepsilon\omega_0 q(1-q) + \omega_1(1-q)^2}{(\omega_0 - \omega_1)q^2 + \omega_1}. \tag{2.13}$$

We want to maximize (2.13) with respect to q where $0 \leq q \leq 1$. Taking the derivative,

$$\begin{aligned}
& \alpha \cdot \frac{\begin{pmatrix} (2\varepsilon\omega_0(1-2q) - 2\omega_1(1-q))\left((\omega_0-\omega_1)q^2+\omega_1\right) \\ \qquad - 2(\omega_0-\omega_1)q\left(2\varepsilon\omega_0 q(1-q)+\omega_1(1-q)^2\right) \end{pmatrix}}{\left((\omega_0-\omega_1)q^2+\omega_1\right)^2} \\
= \quad & \alpha \cdot \frac{2(\omega_0-\omega_1)(\omega_1-\varepsilon\omega_0)q^2 + 2\omega_1(2(\omega_1-\varepsilon\omega_0)-\omega_0)q - 2\omega_1(\omega_1-\varepsilon\omega_0)}{\left((\omega_0-\omega_1)q^2+\omega_1\right)^2}.
\end{aligned}$$

We first evaluate this derivative at the boundaries. At $q = 1$, the numerator is always non-positive with $-2\varepsilon\omega_0^2 \leq 0$. When $q = 0$, the numerator is $-2\omega_1(\omega_1 - \varepsilon\omega_0)$. If this quantity is non-positive, i.e., $\varepsilon \leq \omega_1/\omega_0$, then the coefficient for q^2 given by $2(\omega_0 - \omega_1)(\omega_1 - \varepsilon\omega_0)$ is also non-negative, which implies that (2.13) decreases in q everywhere in the domain $0 \leq q \leq 1$. Thus, if $\varepsilon \leq \omega_1/\omega_0$, we obtain the competitive ratio $1 + \alpha$ by setting $q = 0$ in (2.13).

If $\varepsilon > \omega_1/\omega_0$, the expression in (2.13) attains a maximum in the interior of the domain. The rest of the proof is straightforward calculus. We achieve the maximum given in the statement of the lemma by setting

$$q = \sqrt{\frac{\omega_1}{\omega_0-\omega_1} + \left(\frac{\omega_1}{\omega_0-\omega_1}\cdot\frac{2\varepsilon\omega_0-2\omega_1+\omega_0}{2\varepsilon\omega_0-2\omega_1}\right)^2} - \frac{\omega_1}{\omega_0-\omega_1}\cdot\frac{2\varepsilon\omega_0-2\omega_1+\omega_0}{2\varepsilon\omega_0-2\omega_1}.$$

□

A preemptive policy aggressively searches for high priority jobs by preempting every type 1 job it encounters, completing any low priority residual work only after all jobs are open and all type 0 jobs have completed their processing. Lemma 2 confirms our intuition that this policy performs poorly when prediction error is low. Consider for example an instance that consists exclusively of type 1 jobs. Indiscriminate preemption offers no advantage, as there are no urgent jobs to search for. In this case, a preemptive policy causes on average an α unit of delay in the completion of every job, resulting in a constant $1 + \alpha$

competitive ratio when error rates are small ($\varepsilon \le \omega_1/\omega_0$). Given our assumption that prediction errors are at most one half, we are able to deduce the following corollary.

Corollary 6. *If $\omega_0 < 2\omega_1$, a fully preemptive policy is $(1+\alpha)$-competitive.*

Thus, preemption offers little advantage when the relative weight differential is small.

Lemma 3. *A hybrid policy achieves a competitive ratio of*

$$\mathsf{CR}^{\beta} := 1 + \frac{1}{2}\left(\alpha\varepsilon_1^2 - \lambda + \sqrt{\frac{\omega_0}{\omega_1}\lambda^2 + \frac{\omega_0}{\omega_0-\omega_1}\left(\alpha\varepsilon_1^2\right)^2}\right) \tag{2.14}$$

where

$$\lambda = \varepsilon_0(1+\varepsilon_1) + \frac{\alpha\omega_0}{\omega_0-\omega_1}\varepsilon_1(1-\varepsilon_0) - \frac{\alpha\omega_1}{\omega_0-\omega_1}\varepsilon_1^2. \tag{2.15}$$

Proof. The proof proceeds similarly where C_j^{β} denotes job j's completion time in a hybrid policy. Letting $n_1 = n - n_0$ and $q = n_0/n$,

$$\begin{aligned}
&\frac{\mathbb{E}\left(\sum_{j=1}^{n} w_j C_j^{\beta}\middle| n_0\right)}{\mathsf{OPT}} - 1 \\
&= \frac{\mathbb{E}\left(\alpha\omega_0 X_0 + \alpha\omega_1 Y_0 + (\omega_0-\omega_1)(X-X_0) | n_0\right)}{\mathsf{OPT}} \\
&= \frac{\begin{pmatrix}\alpha\omega_0\varepsilon_1(1-\varepsilon_0)q(1-q)n^2 + \alpha\omega_1\varepsilon_1^2\left((1-q)^2n^2 - (1-q)n\right) \\ \hfill + (\omega_0-\omega_1)\varepsilon_0(1+\varepsilon_1)q(1-q)n^2\end{pmatrix}}{(\omega_0-\omega_1)\left(q^2n^2+qn\right) + \omega_1(n^2+n)}
\end{aligned}$$

then, we approach the limit from below as $n \to \infty$ so the upper bound is

$$\frac{\left(\alpha\omega_0\varepsilon_1(1-\varepsilon_0) + (\omega_0-\omega_1)\varepsilon_0(1+\varepsilon_1)\right)q(1-q) + \alpha\omega_1\varepsilon_1^2(1-q)^2}{(\omega_0-\omega_1)q^2+\omega_1}. \tag{2.16}$$

Using arguments similar to those given in the previous lemma, (2.16) always attains a maximum in the domain $0 \le q \le 1$ when

$$q = \sqrt{\frac{\omega_1}{\omega_0-\omega_1} + \left(\frac{\omega_1}{\omega_0-\omega_1}\cdot\frac{\lambda+\alpha\varepsilon_1^2}{\lambda-\alpha\left(\frac{\omega_1}{\omega_0-\omega_1}\right)\varepsilon_1^2}\right)^2} - \frac{\omega_1}{\omega_0-\omega_1}\cdot\frac{\lambda+\alpha\varepsilon_1^2}{\lambda-\alpha\left(\frac{\omega_1}{\omega_0-\omega_1}\right)\varepsilon_1^2}$$

where λ is as defined above. □

An interpretable upper bound for (2.14) can be derived using the inequality $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$, which yields

$$\mathsf{CR}^{\beta} \leq 1 + \frac{\lambda}{2}\left(\sqrt{\frac{\omega_0}{\omega_1}} - 1\right) + \frac{\alpha\varepsilon_1^2}{2}\left(1 + \sqrt{\frac{\omega_0}{\omega_0 - \omega_1}}\right).$$

Rearranging (2.15), we have

$$\lambda = \varepsilon_0 + \varepsilon_1\left(\alpha + (1-\alpha)\varepsilon_0 + \frac{\alpha\omega_1}{\omega_0 - \omega_1}(1 - \varepsilon_0 - \varepsilon_1)\right). \tag{2.17}$$

We first show that $\lambda/2 \leq \varepsilon$, where ε is the average of ε_0 and ε_1. To do so, it suffices to show that the coefficient to ε_1 in (2.17) is no greater than 1.

$$\begin{aligned}
& 1 - \left(\alpha + (1-\alpha)\varepsilon_0 + \frac{\alpha\omega_1}{\omega_0 - \omega_1}(1 - \varepsilon_0 - \varepsilon_1)\right) \\
= \quad & (1-\alpha)(1-\varepsilon_0) - \frac{\alpha\omega_1}{\omega_0 - \omega_1}(1 - \varepsilon_0 - \varepsilon_1) \\
= \quad & (1-\alpha)\left(1 - \varepsilon_0 - \frac{\alpha}{1-\alpha} \cdot \frac{\omega_1}{\omega_0 - \omega_1}(1 - \varepsilon_0 - \varepsilon_1)\right) \\
= \quad & (1-\alpha)\left(1 - \varepsilon_0 - \beta(1 - \varepsilon_0 - \varepsilon_1)\right) \\
= \quad & (1-\alpha)\left((1-\beta)(1-\varepsilon_0) + \beta\varepsilon_1\right) \geq 0.
\end{aligned}$$

The last inequality follows since every term in the expression is nonnegative, so we have the desired inequality. Recalling that the nonpreemptive competitive ratio is $\mathsf{CR}^{\phi} = 1 + \varepsilon\left(\sqrt{\omega_0/\omega_1} - 1\right)$, we are able to decompose the competitive ratio as follows:

$$\mathsf{CR}^{\beta} \leq 1 + \underbrace{\frac{\lambda}{2}\left(\sqrt{\frac{\omega_0}{\omega_1}} - 1\right)}_{\substack{\leq \mathsf{CR}^{\phi} \\ \text{gains relative to } \mathsf{CR}^{\phi}}} + \underbrace{\frac{\alpha\varepsilon_1^2}{2}\left(1 + \sqrt{\frac{\omega_0}{\omega_0 - \omega_1}}\right)}_{\text{losses relative to } \mathsf{CR}^{\phi}}. \tag{2.18}$$

When the relative urgency $\omega_0/\omega_1 \gg 1$, gains in the β-threshold policy relative to a nonpreemptive policy are large since the multiplier $\sqrt{\omega_0/\omega_1} - 1$ is large. In comparison, the losses are approximately equal to $\alpha\varepsilon_1^2$ and small, which implies a guaranteed performance improvement.

Theorem 4. *The β-threshold policy achieves a competitive ratio*

$$\begin{cases} \mathsf{CR}^{\phi} & \text{if } \rho(1-\beta)\varepsilon_0 + \beta(1-\rho)\varepsilon_1 \geq \min\left(\rho(1-\beta), \beta(1-\rho)\right) \text{ and } \rho \leq \beta, \\ \mathsf{CR}^{\alpha} & \text{if } \rho(1-\beta)\varepsilon_0 + \beta(1-\rho)\varepsilon_1 \geq \min\left(\rho(1-\beta), \beta(1-\rho)\right) \text{ and } \rho > \beta, \text{ and} \\ \mathsf{CR}^{\beta} & \text{if } \rho(1-\beta)\varepsilon_0 + \beta(1-\rho)\varepsilon_1 < \min\left(\rho(1-\beta), \beta(1-\rho)\right). \end{cases}$$

Proof. The theorem combines Lemmas 1, 2 and 3 and the conditions in Corollaries 1, 2 and 3. □

An immediate observation from our competitive analyses is that the worst-case fraction of urgent jobs is inversely proportional to relative priority levels ω_0/ω_1. But more importantly, we are able to characterize how the competitive ratio evolves as a function of prediction error.

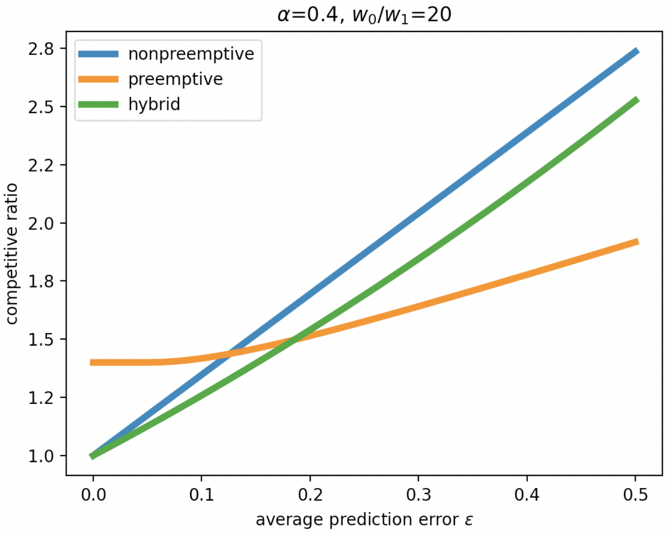


Figure 2.3: Competitive ratios

For analytical purposes, let us assume $\varepsilon = \varepsilon_0 = \varepsilon_1$ and $\omega_0/\omega_1 \gg 1$. We can easily see from our expression of CR^{ϕ} in Lemma 1 that the competitive ratio of a nonpreemptive policy grows linearly in $O(\varepsilon)$, i.e., performance improves with prediction accuracy. In the case of a preemptive policy, the competitive ratio stays constant at $1 + \alpha$ before it starts growing linearly in $O(\alpha\varepsilon)$. Compared with that of a nonpreemptive policy, its growth rate is scaled down by a factor of α where $0 < \alpha < 1$. Finally, Lemma 3 shows that the competitive ratio of a hybrid policy grows *quadratically* in $O\left(\varepsilon^2\right)$. Given $\varepsilon \leq 1/2$, this rate of growth is slower than that of a nonpreemptive policy, offering yet another interpretation of the decomposition of CR^β given in (2.18). These findings are illustrated in Figure 2.3.

Our analyses indicate that, for all three policies that the β-threshold rule admits, performance degrades gracefully as a function of prediction error. As such, we achieve the two qualities that an online algorithm with advice should exhibit: *consistency* and *robustness* [57]. Consistency requires performance improvement when the predictor has low error. The idea is that performance with good advice should be better than performance with poor advice. At the same time, an algorithm should be robust to all inputs, with or without predictions. All three of our policies show improved performance with prediction accuracy. The nonpreemptive and hybrid policies even recover the offline optimum when offered perfect predictions. Our three policies are also robust in that the competitive ratios are bounded above by when error rates are equal to one half. When $\varepsilon = 1/2$, predictions are truly random, i.e., there are no predictions at play.

2.3 Extensions

In this section, we consider a number of extensions to our model that more accurately reflect real-world settings.

2.3.1 Probabilistic Classifiers

We first consider a probabilistic classifier that is able to predict with what probability a job is of high priority. Rather than providing a binary predictive label of urgent vs. non-urgent, this probabilistic classifier directly offers an estimate of p_j. Let us denote these estimated probabilities as $\hat{p}_j$.

We first sort jobs in nonincreasing order of $\hat{p}_j$, breaking ties arbitrarily. As we proceed with our policy, jobs are opened in sorted order. Then, we have the following corollary to Theorem 3.

Corollary 7. *The β-threshold rule is optimal given a probabilistic classifier.*

The β-threshold rule proceeds similarly even with a probabilistic classifier. A preemptive policy is applied to those jobs whose estimated probabilities lie above β, and a nonpreemptive policy is applied to those jobs whose $\hat{p}_j$ values fall below that threshold. The β-threshold rule remains optimal, minimizing the objective across all non-anticipating policies.

What differs from our original model is the measure of error. Beyond binary classification, the false negative and false positive rates ε_0 and ε_1 no longer apply. A more appropriate measure of error in this case would be the logarithmic loss function (also called the cross-entropy loss function) given as follows:

$$\eta = -\frac{1}{n}\sum_{j=1}^{n}(1 - true(j)) \cdot \log\left(\hat{p}_j\right) + true(j) \cdot \log\left(1 - \hat{p}_j\right).$$

The β-threshold rule is the best possible policy for decision-making based on available information that is both imperfect and incomplete. However, its performance depends heavily on the accuracy of the classifier on hand. An exact characterization of performance as a function of the log-loss η remains an open problem.

2.3.2 Probabilistic Learning Outcome

Our model assumes that a radiologist is always able to determine a job's true type at its α-point with probability 1. Perhaps a more realistic model would be to leave some room for doubt. Suppose that at job j's α-point, we learn that job j is an urgent job with some probability $\theta_j \in [0, 1]$. Probability θ_j is a *posterior* probability that offers a better likelihood of job j's urgency based on α units of observed data.

In our notation, $\mathcal{S}$ denotes the set of unopened jobs. Every job in set $\mathcal{S}$ has one unit in remaining work, with an associated a *prior* probability of being an urgent job. These prior probabilities are as defined in (2.1)-(2.2). Let $\mathcal{I}$ denote the set of interrupted, previously preempted jobs that each have $1 - \alpha$ units in residual work. Every job in $\mathcal{I}$ has an associated posterior probability. At every decision point, we make the decision of whether to open a job k with the largest prior probability $p_k = \max_{j \in \mathcal{S}} p_j$, or to complete the remaining $1 - \alpha$ units of work of job i, where job i has the largest posterior probability among all jobs in $\mathcal{I}$ such that $\theta_i = \max_{j \in \mathcal{I}} \theta_j$. For this decision problem, the following modified version of the β-threshold rule is the best possible across all non-anticipating policies.

Theorem 5. *The* $\left(\beta + \frac{\alpha}{1-\alpha} \frac{\omega_0}{\omega_0 - \omega_1} \frac{\theta_i}{1-\theta_i}\right)$*-threshold rule is optimal where* $\theta_i = \max_{j \in \mathcal{I}} \theta_j$*.*

Proof. The proof is nearly identical to the interchange argument given in Theorem 3, with small modifications. The main difference is in recognizing that the low priority job (job i) competing against the next unopened job (job k) in the proof of Theorem 3 is now a low priority job with probability $1 - \theta_i$, and a high priority job with probability θ_i.

As before, we consider the last decision point that deviates from this modified β-threshold rule. If

$$p_k > \beta + \frac{\alpha}{1-\alpha}\frac{\omega_0}{\omega_0 - \omega_1}\frac{\theta_i}{1-\theta_i}$$

and job i is being processed at this decision point, the net change to the objective upon interchange is

$$\underbrace{(-z_0\omega_0(1-\alpha) + \omega_1\left(z\alpha + z_0(1-\alpha)\right))}_{\text{from (2.4)}}(1-\theta_i) + z(\omega_0\alpha)\theta_i$$

where z is the number of jobs opened whose prior probabilities lie above the modified β threshold, and z_0 is the number of type 0 jobs among them. The first half of this expression comes directly from our earlier proof, weighted by the probability that job i is a low priority job. If job i is an urgent job with weight ω_0, it incurs an α unit of delay in completion for each of the z jobs opened during interchange. Then,

$$\begin{aligned} & \left(-z_0\omega_0(1-\alpha) + \omega_1\left(z\alpha + z_0(1-\alpha)\right)\right)(1-\theta_i) + z(\omega_0\alpha)\theta_i \\ = \quad & -(\omega_0-\omega_1)(1-\alpha)(1-\theta_i)\left(z_0 - \underbrace{\left(\beta + \frac{\alpha}{1-\alpha}\frac{\omega_0}{\omega_0-\omega_1}\frac{\theta_i}{1-\theta_i}\right)}_{\text{modified } \beta} z\right). \end{aligned}$$

In expectation, the overall change to the objective is negative since each of the z jobs have prior probabilities that lie above the modified β threshold.

The second case uses an identical argument. We modify (2.5) to account for the possibility that job i is a type 0 job. Then,

$$\underbrace{(-\ell\omega_1 + w_k\ell(1-\alpha))}_{\text{from (2.5)}}(1-\theta_i) - \ell(\omega_0\alpha)\theta_i$$

and the rest of the proof proceeds similarly. □

We recover the original β-threshold rule when $\theta_i = 0$, which is equivalent to learning that job i is a non-urgent job with probability 1. This added uncertainty raises the β threshold bar for opening new jobs to account for the possibility that job i is an urgent job.

2.3.3 Job Arrivals Over Time

We had previously assumed that all jobs are available for processing at time 0. In this section, we consider the case where jobs arrive over time and are released for processing at various points in time. Each job j has an associated release date $r_j \geq 0$ and cannot be processed before then. At any given time, we assume no knowledge of jobs arriving in the future.

The offline version of this problem in which jobs' true types are known a priori can be written as $1|r_j, p_j = 1, pmtn|\sum\{\omega_0, \omega_1\}C_j$ in the scheduling notation of Graham et al. [40]. The following theorem identifies an optimal policy for this offline problem.

Theorem 6. *The weighted shortest remaining processing time (WSRPT) rule is an optimal policy for* $1|r_j, p_j = 1, pmtn|\sum\{\omega_0, \omega_1\}C_j$.

Proof. Consider an optimal schedule where job k is being processed at time t.

Suppose there exists another available job j at t such that

$$\frac{w_k}{x_k(t)} < \frac{w_j}{x_j(t)} \tag{2.19}$$

where $x_j(t)$ denotes the amount of work remaining in job j at time t. If $w_j = w_k$, the optimal policy is in violation of the SRPT rule so we immediately have a contradiction [74]. We therefore assume that $w_j \neq w_k$. In addition, we also assume without loss of generality that job j is the job with the largest weight-to-remaining-work ratio among all available jobs at t. We establish a contradiction by interchange.

Despite our assumption that the optimal schedule prioritizes job k over job j at time t, we do not know whether job k was actually completed prior to job j. We let C_j and C_k denote the completion times of jobs j and k in the optimal schedule, respectively, and consider both cases.

i $C_j < C_k$: we use a pairwise interchange argument similar to that used in the proof of optimality of SRPT. Starting from t, we take the first $x_j(t)$ units devoted to processing jobs j or k in the optimal schedule, and use that time to process job j to completion at $\hat{C}_j$. The remaining $x_k(t)$ units of time are then used to process job k with completion time $\hat{C}_k = C_k$. This interchange only affects the completion time of job j, and $\hat{C}_j < C_j$ by construction, so we have our desired contradiction.

ii $C_k < C_j$: in this case, we require some additional pieces that are unique to our problem with two distinct weights. We first claim that job j is the only job of weight w_j that is processed in the interval $[t, C_j)$. At time t, job j has the largest weight-to-remaining-work ratio, so it would be against the SRPT rule to process any other available job of weight w_j until job j is complete.

The same is true of any job of weight w_j released in the interval $[t, C_j)$ since $x_j(t) \leq 1$ and every newly arriving job has 1 unit of remaining work. Using a similar argument for job k in the interval $[t, C_k)$, we can conclude that only jobs j and k are processed in $[t, \min(C_k, C_j)) = [t, C_k)$.

It is possible, however, that other jobs are processed in $[C_k, C_j)$. By our earlier claim, only jobs of weight w_k can be processed in this interval. Let $\mathcal{A}$ denote the set of jobs processed in $[C_k, C_j)$ where, for every job $\ell \in \mathcal{A}$, $w_\ell = w_k$ holds. We claim that every job $\ell \in \mathcal{A}$ satisfies

$$\frac{w_\ell}{x_\ell(t)} < \frac{w_j}{x_j(t)}. \tag{2.20}$$

For notational convenience, we shall continue to use $x_\ell(t)$ on jobs that are released after t, as we can simply set $x_\ell(t) = 1$ without affecting the analysis. If job ℓ has release date $r_\ell \leq t$, then $x_\ell(t) \geq x_k(t)$ since the optimal schedule would otherwise be in violation of the SRPT rule by processing job k instead of ℓ at time t. Combined with (2.19), we obtain the inequality. The same is true if job ℓ has release date $r_\ell \in [t, C_j)$ since $x_k(t) \leq x_\ell(t) = 1$.

Lastly, we argue that every job $\ell \in \mathcal{A}$ has completion time $C_\ell \in [C_k, C_j)$. Suppose on the contrary that there exists a job that is partially processed in $[C_k, C_j)$ that completes sometime after C_j. Then, shifting the time units devoted to processing this job to the end of the $[C_k, C_j)$ interval allows job j to be completed earlier without affecting the completion time of any other job. Doing so strictly improves the objective and contradicts the fact that we have an optimal schedule.

We finally have all the ingredients we need to proceed with the interchange. We first process job j in the first $x_j(t)$ units of $[t, C_j)$, followed by jobs in $\mathcal{A} \cup \{k\}$ in the remainder of the interval $[t + x_j(t), C_j)$. Then, for each job in

$\mathcal{A} \cup \{k\}$, there is a delay in completion of at most $x_j(t)$ units. Job j, on the other hand, completes $x_k(t) + \sum_{\ell \in \mathcal{A}} x_\ell(t)$ units earlier in the schedule. The net effect to the objective is thus bounded above by

$$\begin{aligned}
& - w_j \left(x_k(t) + \sum_{\ell \in \mathcal{A}} x_\ell(t) \right) + |\mathcal{A} \cup \{k\}| w_k x_j(t) \\
= & - w_j x_k(t) + w_k x_j(t) - w_j \left(\sum_{\ell \in \mathcal{A}} x_\ell(t) \right) + |\mathcal{A}| w_k x_j(t) \\
= & \underbrace{-w_j x_k(t) + w_k x_j(t)}_{< 0 \text{ by } (2.19)} + \sum_{\ell \in \mathcal{A}} \underbrace{(-w_j x_\ell(t) + w_\ell x_j(t))}_{< 0 \text{ by } (2.20)} < 0
\end{aligned}$$

which contradicts the fact that we have an optimal schedule. □

It is worth adding that the theorem above does not generalize to problems of the same setting with three or more distinct weights. In particular, given our assumptions in (2.19), our interchange argument relies on job j being the job with the largest weight-to-remaining-work ratio among all jobs completing in the interval $[t, C_j)$. We have shown with (2.20) that this condition always holds when there are two distinct weights. When there are three or more distinct weights, we can easily construct examples for which this condition no longer holds, for example, by scheduling the arrival of a job with very large weight in $[t, C_j)$.

The WSRPT rule combines two well-known scheduling results: the WSPT rule (Theorem 1) and the SRPT rule (Theorem 2). The WSRPT rule itself is not new; it has been used in other works as a popular heuristic (see [13, 87], for example). Nevertheless, to our knowledge, Theorem 6 is the first result on WSRPT optimality, and $1|r_j, p_j = 1, pmtn| \sum \{\omega_0, \omega_1\} C_j$ is the first scheduling problem for which WSRPT is shown to be optimal.

Competitive Analysis We now consider the β-threshold rule when jobs arrive into the system over time. At each decision point, we make decisions based on an updated set of unopened jobs that accounts for any new job arrivals since our last decision point. These newly added jobs enter the queue according to their predicted priorities. Whereas our hybrid policy previously allowed a one-time switch from a preemptive policy to a nonpreemptive policy, the arrival of a high priority job could trigger preemptions when necessary, resulting in alternating preemptive and nonpreemptive regimes.

Online job arrivals add yet another layer of randomness and complexity to our model. Our efforts in competitive analysis incorporating both job arrivals and imperfect predictions were not yet fruitful. In what follows, we present our results when job arrivals are present with perfect type predictions.

Let OPT denote the offline optimum obtained by WSRPT, and let ALG_0 denote the performance of the online β-threshold policy when job priorities are known a priori. The main difference between these two policies under consideration is that we are able to preempt a job whenever necessary in OPT, but may do so at most once at a job's α-point in ALG.

The online policy ALG_0 assumes that true job priorities are given to us at time of job arrival. This is a deterministic online problem where preemptions are limited to α-points, so we might express this problem as $1|r_j, p_j = 1, \alpha\text{-}pmtn| \sum\{\omega_0, \omega_1\}C_j$ in the scheduling notation of Graham et al. [40]. We follow the β-threshold rule at each decision point, where our set of unopened jobs includes jobs that have arrived since our last decision point. We review each decision in detail to highlight that each of our decisions are consistent with WS-RPT. First, the existence of any unprocessed type 0 job will trigger a preemption at an α-point. Preempting a type 1 job at an α-point is WSRPT-consistent since, by Assumption 1,

$$\omega_1 < \omega_0(1-\alpha) \iff \frac{\omega_1}{1-\alpha} < \frac{\omega_0}{1}.$$

Type 0 jobs will then complete nonpreemptively. When only type 1 jobs remain, any partially processed type 1 job will be processed to completion before we move on to an unopened type 1 job. This is consistent with the β-threshold rule, the SRPT rule, and by extension, the WSRPT rule. Thus, when true job types are known a priori, ALG_0 is an optimal policy for $1|r_j, p_j = 1, \alpha\text{-}pmtn| \sum\{\omega_0, \omega_1\}C_j$. We now compare its performance against OPT. Our proofs frequently rely on the following inequality, widely known as the mediant inequality.

The Mediant Inequality. For any positive real numbers $a, b, c, d > 0$,

$$\frac{a+b}{c+d} \leq \max\left(\frac{a}{c}, \frac{b}{d}\right).$$

Theorem 7. ALG_0 *is* $\max\left(1+\alpha, \frac{2}{1+\alpha}\right)$*-competitive, and* $\sqrt{2}$*-competitive if we choose* $\alpha = \sqrt{2} - 1$.

Proof. We proceed by running the online β-threshold policy and WSRPT in parallel. Both policies schedule the same set of n jobs arriving over time, where true job priorities are immediately observable upon job arrival. We refer to the schedule generated by WSRPT as the optimal schedule.

We first discuss some reasonable assumptions we can impose on the data. Without loss of generality, we assume $\min_j r_j = 0$, and that there is at least one job of each type in the dataset. We also assume that each of these n jobs are processed without idle time in the optimal schedule so that the last job completes at time n. To see why, first observe that both policies are work-conserving. Any dataset that prompts a machine to become idle in an optimal schedule will simultaneously create idle time in a schedule generated by our online policy. Let us partition the dataset whenever there is idle time. Because our objective functions are linear, we can apply the mediant inequality to the competitive ratio based on said partition. Thus it suffices to consider a set of jobs that does not generate idle time.

We proceed by identifying ways to further partition our set of jobs until we have a minimal set of jobs that gives the worst case performance. In order to do so, we need the following claims.

Claim 1. *Type 1 jobs begin processing at the same time in* OPT *and in* ALG_0*. This start time is always integer.*

Proof of Claim 1. Let $t > 0$ be any time at which a type 1 job begins its processing in an optimal schedule. By the optimality of the WSRPT rule, every type 0 job released prior to t has completed by t, and no type 1 job that has begun its processing prior to t is left unfinished. Integrality of t follows naturally.

Type 1 jobs also start at integer time points in the schedule generated by our online β-threshold policy because the policy requires that any partially processed jobs be completed before opening a new type 1 job. Consider time t as defined above. The optimal schedule implies that all type 0 jobs released prior to t have release dates no later than $t-1$. Because there is at least one decision point in the interval $[t-1, t)$, any type 0 job released prior to t must have completed by t in the schedule generated by the online policy. Then, by our assumption that precludes any idle time in the schedule, integer units of work have been done on type 1 jobs by t and the result follows. □

Claim 1 allows us to partition the schedule whenever a type 1 job begins its processing. Within each partitioned block, the same set of jobs will have completed processing in both policies. Thus, by the mediant inequality, we consider one such block. This implies that it suffices to consider a dataset with exactly one type 1 job. We call this job k. Without loss of generality, we assume that job k begins its processing at time 0. Let C_k^* and C_k^0 denote the completion time of job k in OPT and ALG_0, respectively.

Claim 2. $C_k^* \geq C_k^0$.

Proof of Claim 2. First observe that C_k^* and C_k^0 are positive integers for the same reasons given in the proof of Claim 1. Suppose on the contrary that $C_k^* < C_k^0$. Then there exists some job $\ell \neq k$ of type 0 that is *not* processed prior to C_k^* in the optimal schedule that is being processed in ALG_0 at time C_k^*. Preemptions only occur at α-points in ALG_0, so job ℓ must have begun its processing at $C_k^* - (1-\alpha)$, which implies that $r_\ell \leq C_k^* - (1-\alpha)$. By Assumption 1, it follows that $r_\ell \leq C_k^* - (1-\alpha) < C_k^* - (\omega_1/\omega_0)$.

The optimal schedule follows WSRPT, so delaying the processing of any type 0 job in favor of completing job k would occur only if a type 0 job arrives at such a time that the remaining work in job k, x_k, satisfies $\omega_1/x_k > \omega_0/1 \iff x_k < \omega_1/\omega_0$. Said differently, only those type 0 jobs arriving after time $C_k^* - (\omega_1/\omega_0)$ would be processed outside of the $[0, C_k^*)$ interval in an optimal schedule, and by Assumption 1, also outside of $[0, C_k^*)$ in ALG_0. Since $r_\ell < C_k^* - (\omega_1/\omega_0)$, job ℓ should have completed before job k in an optimal schedule, which establishes the desired contradiction. □

An important byproduct of the proof of Claim 2 is that every job that completes in the interval $[0, C_k^*)$ in $\mathsf{ALG_0}$ also completes within the same interval in an optimal schedule. Both policies are work-conserving, so the converse also holds. Let $\mathcal{A}$ denote the set of jobs completing in this interval. Then, by another application of the mediant inequality, it suffices to consider the set of jobs $\mathcal{A}$, where job k is the only type 1 job therein. An immediate consequence of this is an upper bound of $\max(\alpha, 1-\alpha)$ on the delay in type 0 job completion times in $\mathsf{ALG_0}$ relative to those in OPT. Intuitively, this bound captures how long a type 0 job will have to wait until the next decision point while job k is being processed. Thus,

$$\begin{aligned}
\frac{\mathsf{ALG_0}}{\mathsf{OPT}} &= \frac{\omega_1 C_k^0 + \sum_{j\in\mathcal{A}\setminus\{k\}} \omega_0 C_j^0}{\omega_1 C_k^* + \sum_{j\in\mathcal{A}\setminus\{k\}} \omega_0 C_j^*} \\
&\le \frac{\sum_{j\in\mathcal{A}\setminus\{k\}} \cancel{\omega_0} C_j^0}{\sum_{j\in\mathcal{A}\setminus\{k\}} \cancel{\omega_0} C_j^*} && \text{by the mediant inequality, since } C_k^* \ge C_k^0 \\
&\le \max_{j\in\mathcal{A}\setminus\{k\}} \left(\frac{C_j^0}{C_j^*}\right) && \text{by the mediant inequality} \\
&= \max_{j\in\mathcal{A}\setminus\{k\}} \left(\frac{C_j^* + \delta_j}{C_j^*}\right)
\end{aligned}$$

where δ_j is job j's delay in completion in $\mathsf{ALG_0}$ relative to OPT. Finally, using $\delta_j \le \max(\alpha, 1-\alpha)$ and finding the earliest possible completion time in the optimal schedule for each type of delay,

$$\begin{aligned}
\max_{j\in\mathcal{A}\setminus\{k\}} \left(\frac{C_j^* + \delta_j}{C_j^*}\right) &\le \max\left(1 + \frac{\alpha}{1}, 1 + \frac{1-\alpha}{1+\alpha}\right) \\
&= \max\left(1+\alpha, \frac{2}{1+\alpha}\right),
\end{aligned}$$

which proves the result. □

Our results offer some guidance as to which values of α might be effective when dealing with limited preemption points. But given job arrivals and perfect predictions, the β-threshold rule is 2-competitive regardless of the α value that we choose. Given imperfect predictions without job arrivals, on the other hand, we obtain our earlier result given in Theorem 4. Competitive analysis featuring both uncertainties remains an open problem.

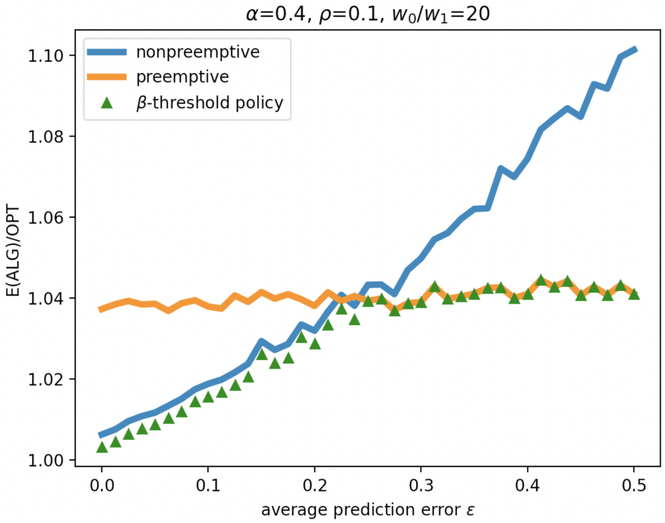


Figure 2.4: Expected performance when jobs arrive over time

Despite the lack of theoretical guarantees, empirical evaluations of the β-threshold rule under realistic job arrival scenarios and imperfect prediction show that our policy still performs very well. Figure 2.4 plots the expected performance of the β-threshold rule as a function of prediction error, where performance is normalized by the offline optimum obtained by WSRPT. For illustrative purposes, we assume $\varepsilon_0 = \varepsilon_1$. In this plot, we assume that jobs are arriving according to a Poisson arrival process with mean interarrival time 0.9 (given unit processing times). The figure shows that our policies exhibit near-optimal performance, and that our β-threshold rule of alternating between the nonpreemptive and preemptive regimes outperforms both non-adaptive policies when we are given high quality advice.

Our experiments thus far reveal that our original stylized model without job arrivals results in the worst-case performance. This is surprising to us, and also somewhat counterintuitive given classic results in scheduling theory involving job release dates. One possible explanation for this could be the higher opportunity costs of misprediction stemming from having a long line of jobs waiting in the queue, but truthfully, we do not have a good answer for this yet.

2.4 Discussion and Future Directions

The work presented in this chapter was motivated by recent interest in using machine learning algorithms for patient triage and prioritization. We modeled this as a learning-augmented online scheduling problem in which we are given good but imperfect predictions of patient risk, and sought to capture the tradeoff between the need to prioritize emergency cases and the potential costs of misprediction. We presented a simple threshold-based policy that addressed these concerns and proved that our policy is in fact the best possible in certain stylized settings. The policy was also shown to remain effective in more realistic settings.

The model that we studied is grounded in reality. For many radiologists, preemptions and interruptions are simply facts of life, as is the fact that they are trained to collect information in real time while processing each patient case. In that sense, our policy recommendation is intuitive and easy to implement, and more importantly, does not require an overhaul of existing systems and Modality Worklists that are already in place. While it would be impossible to implement the β-threshold rule by the book in a clinical setting, we do believe that our policy can offer qualitative guidance on how to think about and respond to predictions of patient risk in connection with other input parameters.

That said, our work in this area is far from complete. Several concrete next steps have been outlined in Section 2.3, including exact characterizations of performance with probabilistic classifiers or with probabilistic learning outcomes. Theoretical guarantees of performance of the β-threshold rule with online job arrivals also remain an open problem.

Even beyond these extensions, there are many interesting directions that we can explore for future research. One natural direction would be to generalize our stylized model by allowing granularity in prioritization schemes beyond a binary classification of urgent vs. non-urgent. From a practical perspective, clinics tend to have their own internal methods of categorizing urgency levels. For example, the Department of Radiology at the Weill Cornell Medical Center categorizes urgency levels by the following:

- Critical (JCAHO[1]-designated): immediate communication required
- Emergent: immediate communication required
- Urgent: communication required in under 4 hours
- Important: closed-loop communication required but not in an urgent time frame (1-2 week limit).

While this is clearly a natural next step to consider, it is less evident whether our optimal policy structure extends under this more general setting. We have observed, for example, that the optimality of the WSRPT rule breaks immediately upon adding a third priority class.

[1] Joint Commission on Accreditation of Healthcare Organizations

Another direction would be to consider various preemptive strategies that better reflect clinical scenarios. In our scheduling formulation, preemptions could be used to model the many different ways in which radiologists learn true job types over time. One extension might be to consider multiple α-points of preemption. For example, given $0 < \alpha_1 < \alpha_2 < \cdots < 1$, we might imagine radiologists having improved confidence about a job's true priority with additional time spent processing that job. We could also consider varying preemption points for each job, for instance by letting job j preempt at a unique α_j-point once a radiologist meets a certain level of confidence. It would then be interesting to observe how performance evolves as a function of these preemption confidence levels.

In a similar vein, it is often the case that preemption comes at a cost. Our model assumes that the work required to complete an interrupted job is exactly the same as if it had not been interrupted. Realistically, it might take a while for a radiologist to warm up to a job, in which case restarting a previously preempted job may require an extra factor of $\gamma > 1$ in processing time. Early attempts at tackling this problem with friction costs have not been successful due to difficulties in having to differentiate decision points by continuity in job processing.

Continued advances in machine learning techniques mean that, over time, algorithms will likely become better at detecting abnormalities in medical images. Our current model assumes fixed error rates based on guarantees on expected generalization error, but we could also consider applying Bayesian inference techniques to update error rates over time based on observed data. This might lead to an adaptive β-threshold policy for which we might seek convergence results.

Finally, in the spirit of scheduling research, we could consider how the policy performs when there are multiple radiologists, i.e., parallel machines.

CHAPTER 3

SPT OPTIMALITY VIA LINEAR PROGRAMMING

Consider the problem of scheduling jobs on identical parallel machines to minimize average job completion times. Each job j is available at time 0 and requires $p_j > 0$ units in uninterrupted processing time. Each machine can only process one job at a time. Letting C_j denote the completion time of job j, this problem is $P||\sum C_j$ in the notation of Graham et al. [40]. One of the oldest and most widely known results in scheduling theory is that this problem is solvable in polynomial time [20]. An optimal schedule can be constructed by the Shortest Processing Time (SPT) rule that begins processing a job not yet processed with the shortest processing time whenever a machine is idle.

We present a new proof of correctness of SPT via linear programming (LP). We use an LP formulation previously introduced by Balas [8] and further developed by Wolsey [86], Queyranne [70], Queyranne and Wang [72], Schulz [75] and Hall et al. [42]. Earlier proofs of correctness of the SPT rule rely on *coefficient matching* (see Brucker [17], Lawler et al. [52], and Lenstra and Shmoys [53], for example), but to the best of our knowledge, this is the first LP-based proof.

Our proof of correctness of SPT uses a second scheduling problem that involves job weights $w_j > 0$ for each job j. The general problem $P||\sum w_jC_j$ is NP-hard [52]. One reaction to this NP-hardness result is that an LP-based proof for $P||\sum C_j$ (and the associated structural results then implied by LP duality) should not be possible. However, some special cases of the weighted problem $P||\sum w_jC_j$ are known to be polynomial-time solvable, and in fact, one such special case has an equivalence with our main problem $P||\sum C_j$. This equivalent weighted problem comes with strong structural properties that we are able to exploit using LP techniques. The resulting LP solution is then transformed into an optimal solution for $P||\sum C_j$, which gives an alternate LP-based proof of correctness of SPT.

Identifying an appropriate weighted variant is a critical step in our proof. Our methods generalize a single-machine result based on a 1997 observation by Hall and Chudak [41], and are based on geometric insights from two-dimensional Gantt charts. Gantt charts have already proven useful for tackling various scheduling problems [75, 42, 35], so we expect our methods to also find further uses. To demonstrate this, we apply the same principles in more generalized settings in the last section of this chapter.

The remainder of this chapter is organized as follows. In Section 3.1, we discuss the geometric insights from two-dimensional Gantt charts that reveal a related scheduling problem. Linear programming methods are used to solve this problem and establish SPT optimality in Section 3.2. Section 3.3 extends the idea in uniform and unrelated parallel machine settings.

3.1 Insights from Two-Dimensional Gantt Charts

Gantt charts are useful for visualizing schedules over time, especially for a single machine. Traditional Gantt charts are unidimensional in time. In a nonpreemptive schedule, a machine may block off p_j units in uninterrupted processing time for job j. If job j begins processing at time t, then its completion time $C_j = t + p_j$. See Figure 3.1.

We introduce job weights in two-dimensional Gantt charts. With time on the horizontal axis and the total remaining unprocessed job weight on the vertical

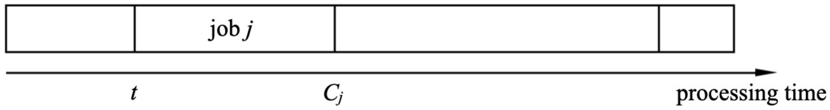


Figure 3.1: A one-dimensional Gantt chart

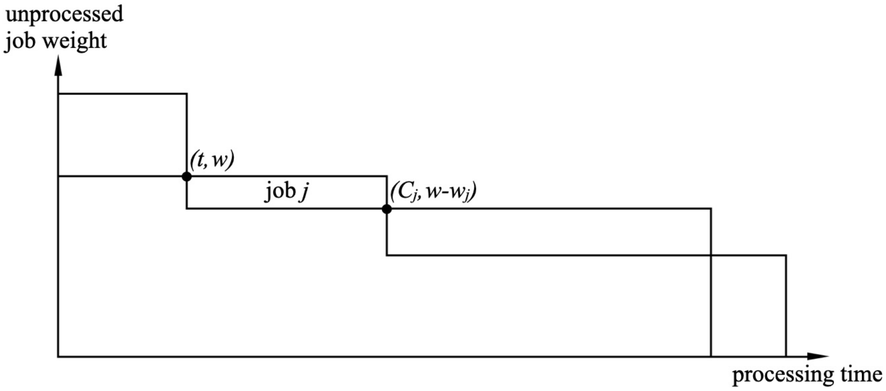


Figure 3.2: A two-dimensional Gantt chart

axis, job j with weight w_j is represented as a rectangular block of width p_j and height w_j. When job j begins processing with coordinate (t, w) on its upper-left corner, it completes with coordinate $(C_j, w - w_j)$ on its lower-right corner as illustrated in Figure 3.2. Letting $\mathcal{N}$ denote the set of all jobs in a schedule, $\sum_{j\in\mathcal{N}} w_jC_j$ is the area under the curve in a two-dimensional Gantt chart. Solving the single-machine problem $1||\sum w_jC_j$ is therefore equivalent to finding a sequence of jobs that minimizes this area under the curve in a two-dimensional Gantt chart.

Two-dimensional Gantt charts have been widely explored for various single-machine problems, as shown in Hall et al. [42], Schulz [75], and Goemans and Williamson [35], for example. In comparison, their applications in parallel-machine settings are relatively limited (a notable exception is the paper by Eastman et al. [29] that introduced the concept of two-dimensional Gantt charts in 1964). Parallel-machine schedules are difficult to interpret graphically when multiple jobs of varying widths are being processed at the same time, each on a different machine.

Two observations are used to transform a two-dimensional Gantt chart for $P||\sum C_j$ for better interpretability. This procedure will also reveal a second scheduling problem that has an equivalence relation with $P||\sum C_j$. As will become clear shortly, the equal-weighted nature of our objective is a key feature in this process.

The first observation is that any feasible parallel-machine schedule with m machines may be decomposed into m feasible single-machine schedules without altering the objective value. Of course, the converse is also true when the set of jobs $\mathcal{N}$ is partitioned into m disjoint sets of jobs.

The next insight is due to Hall and Chudak's 1997 observation that reflecting any feasible, bounded two-dimensional Gantt chart over the identity line preserves the objective value [41]. More precisely, suppose there is a feasible schedule for $1||\sum w_j C_j$ where jobs are described by the set of parameters $\{(p_j, w_j) : j \in \mathcal{N}\}$. Then, we can construct an instance that shares the same area under the curve by scheduling in *reverse order* the set of jobs described by $\{(\hat{p}_j, \hat{w}_j) : j \in \mathcal{N}\}$, where $\hat{p}_j = w_j$ and $\hat{w}_j = p_j$ for each j. See Figure 3.3 for an illustration.

Now consider any feasible schedule for $P||\sum C_j$ with job inputs $\{(p_j, w_j) : j \in \mathcal{N}\}$. In the absence of weights, let $w_j = p$ where p is some constant. We decompose this schedule into m single-machine schedules and flip the weights and processing times of each job according to the Hall and Chudak observation. Doing so creates a set of jobs with *equal processing times* and general weights described by $\{(\hat{p}_j, \hat{w}_j) : j \in \mathcal{N}\}$, where $\hat{p}_j = w_j = p$ and $\hat{w}_j = p_j$, and reverses the

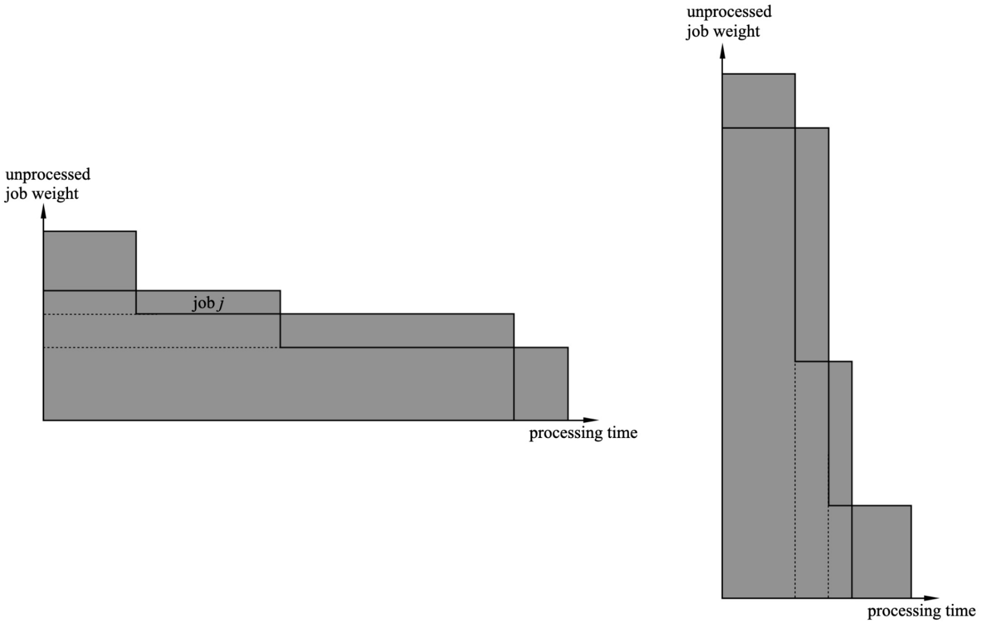


Figure 3.3: An illustration of Hall and Chudak's 1997 observation

order in which jobs are processed on each machine. Putting these m newly created single-machine schedules together, we obtain a feasible parallel-machine schedule for the problem $P|p_j = p|\sum w_jC_j$: an equal-processing-time variant of the weighted problem. The schedules for $P||\sum C_j$ and $P|p_j = p|\sum w_jC_j$ share the same objective value.

This bijection between an input to $P||\sum C_j$ and what we shall call a *flipped input* to $P|p_j = p|\sum w_jC_j$ implies that solving one solves the other. Between the two, $P|p_j = p|\sum w_jC_j$ is a much more attractive problem to solve given its equal processing time structure. Since all jobs are available at time 0 and require p units in processing time, job completion times are always at integer multiples of p in an optimal schedule. A two-dimensional Gantt chart for $P|p_j = p|\sum w_jC_j$ therefore reads like that of a single-machine problem in which each job is a collection of at most m jobs of equal width p. Sequencing jobs in order of nonincreasing w_j/p_j is optimal for the single-machine problem $1||\sum w_jC_j$

by Smith's Weighted Shortest Processing Time (WSPT) rule [78]. By extension, sorting jobs in order of nonincreasing w_j and sequencing collections of m jobs in sorted order is optimal for $P|p_j = p|\sum w_jC_j$. We give a formal proof of this in the following section.

3.2 SPT Optimality

Consider the following linear program for $P|p_j = p|\sum w_jC_j$, which refines the frameworks of Wolsey [86] and Queyranne [70].

$$\min \sum_{j\in\mathcal{N}} w_jC_j \tag{3.1}$$

$$\text{s.t.} \sum_{j\in\mathcal{S}} C_j \geq f(\mathcal{S}) \quad \text{for all } \mathcal{S} \subseteq \mathcal{N}, \tag{3.2}$$

where

$$f(\mathcal{S}) = \frac{p}{2}\left(\left\lceil \frac{|\mathcal{S}|}{m} \right\rceil^2 \cdot (|\mathcal{S}| \bmod m) + \left\lfloor \frac{|\mathcal{S}|}{m} \right\rfloor^2 \cdot (m - |\mathcal{S}| \bmod m) + |\mathcal{S}|\right).$$

The derivation for the functional form of $f(\mathcal{S})$ builds on earlier works that describe the convex hull of feasible completion time vectors. Balas [8], Wolsey [86], Queyranne and Wang [72], Queyranne [70], and Queyranne and Schulz [71] have extensively studied scheduling polyhedra for single machines. Of particular interest are the valid inequalities

$$\sum_{j\in\mathcal{S}} p_jC_j \geq \frac{1}{2}\left(\sum_{j\in\mathcal{S}} p_j\right)^2 + \frac{1}{2}\sum_{j\in\mathcal{S}} p_j^2 \quad \text{for all } \mathcal{S} \subseteq \mathcal{N} \tag{3.3}$$

that capture all permutations of completion times as shown by Wolsey [86] and Queyranne [70]. Valid inequalities have also been derived for parallel machines and subsequently tightened by Schulz [75] and Hall et al. [42]. We expand on

Schulz's 1996 derivation [75] to obtain tighter inequalities given equal processing times. For every $\mathcal{S} \subseteq \mathcal{N}$, consider any partition $\mathcal{S} = \mathcal{S}_1 \cup \mathcal{S}_2 \cup \cdots \cup \mathcal{S}_m$:

$$\begin{aligned} \sum_{j \in \mathcal{S}} p_j C_j &= \sum_{i=1}^{m} \sum_{j \in \mathcal{S}_i} p_j C_j \\ &\geq \sum_{i=1}^{m} \left\{ \frac{1}{2} \left(\sum_{j \in \mathcal{S}_i} p_j \right)^2 + \frac{1}{2} \sum_{j \in \mathcal{S}_i} p_j^2 \right\} \qquad \text{by (3.3)} \\ &= \frac{1}{2} \left\{ \sum_{i=1}^{m} \left(\sum_{j \in \mathcal{S}_i} p_j \right)^2 + \sum_{j \in \mathcal{S}} p_j^2 \right\}. \end{aligned}$$

Given $p_j = p$, it follows that

$$\sum_{j \in \mathcal{S}} C_j \geq \frac{p}{2} \left(\sum_{i=1}^{m} |\mathcal{S}_i|^2 + |\mathcal{S}| \right).$$

Here, we exploit the fact that set cardinalities are integral: the right-hand side is minimized when the number of jobs assigned to each machine is as balanced as possible. That is, given $|\mathcal{S}| = am + b$ where $a, b \in \mathbb{Z}_+$ and $b < m$, then b machines will be assigned $a + 1$ jobs and the remaining $m - b$ machines will be assigned a jobs. Thus, the following inequalities remain valid for $P|p_j = p|\sum w_j C_j$, resulting in (3.2):

$$\sum_{j \in \mathcal{S}} C_j \geq \frac{p}{2} \left(\left\lceil \frac{|\mathcal{S}|}{m} \right\rceil^2 \cdot (|\mathcal{S}| \bmod m) + \left\lfloor \frac{|\mathcal{S}|}{m} \right\rfloor^2 \cdot (m - |\mathcal{S}| \bmod m) + |\mathcal{S}| \right)$$

for every $\mathcal{S} \subseteq \mathcal{N}$, where $\lfloor x \rfloor$ is the largest integer less than or equal to x and $\lceil x \rceil$ is the smallest integer greater than or equal to x.

Let $\mathcal{P}$ be the polyhedron defined by the valid inequalities in (3.2).

Lemma 4. *$\mathcal{P}$ is a supermodular polyhedron with integer vertices.*

Proof. First, we show that the set function f is integer-valued. For any $|\mathcal{S}| =$

$am + b$ such that $a, b \in \mathbb{Z}_+$ and $b < m$,

$$\begin{aligned} f(\mathcal{S}) &= \frac{p}{2}\left(\left\lceil \frac{|\mathcal{S}|}{m} \right\rceil^2 \cdot (|\mathcal{S}| \bmod m) + \left\lfloor \frac{|\mathcal{S}|}{m} \right\rfloor^2 \cdot (m - |\mathcal{S}| \bmod m) + |\mathcal{S}|\right) \\ &= \frac{p}{2}\left((a+1)^2 \cdot b + a^2(m-b) + am + b\right) \\ &= \frac{p}{2}\left(a^2 m + am + 2ab + 2b\right) = \frac{p}{2}\left(a(a+1)m + 2(a+1)b\right) \end{aligned}$$

which establishes integrality.

Next, we show that f is supermodular. First observe that $f(\emptyset) = 0$. By definition, a function is supermodular if, $\forall A \subseteq B \subseteq \mathcal{N}$ and $i \notin B$,

$$f(B \cup \{i\}) - f(B) \geq f(A \cup \{i\}) - f(A).$$

Suppose that $|A| = a_1 m + a_2$ and $|B| = b_1 m + b_2$, where $a_1, a_2, b_1, b_2 \in \mathbb{Z}_+$ and $a_2, b_2 < m$. Since $A \subseteq B$, we know $a_1 \leq b_1$. Notice that $f(A \cup \{i\})$, compared to $f(A)$, will place one additional job into a machine with a_1 jobs. All other machines remain unaffected. Therefore,

$$\begin{aligned} f(A \cup \{i\}) - f(A) &= \frac{p}{2}\left((a_1+1)^2 - a_1^2 + 1\right) \\ &= p\left(a_1 + 1\right) \end{aligned}$$

and similarly,

$$f(B \cup \{i\}) - f(B) = p\left(b_1 + 1\right).$$

By assumption, $a_1 \leq b_1$ and so the result follows. □

Theorem 8. *The valid inequalities in (3.2) completely describe the scheduling polyhedron for $P|p_j = p|\sum_j w_j C_j$.*

Proof. Since we already know the validity of the inequalities in question, what remains to show is that $\mathcal{P}$ is contained in the scheduling polyhedron for $P|p_j =$

$p|\sum_j w_j C_j$. Let C^* be an arbitrary vertex of $\mathcal{P}$, and let w be a vector such that C^* is the unique solution to the linear program $\min\{w^\intercal C : C \in \mathcal{P}\}$. Without loss of generality, suppose jobs are sorted in nonincreasing order of weights. Given Lemma 4, the greedy algorithm for supermodular polyhedra implies that

$$\begin{aligned} C_j^* &= f\left(\{1, \ldots, j\}\right) - f\left(\{1, \ldots, j-1\}\right) \\ &= p\left(\left\lfloor \frac{j-1}{m} \right\rfloor + 1\right) \end{aligned} \tag{3.4}$$

for all $j \in \mathcal{N}$. Given our equal-processing-time assumption, all completion times must occur at multiples of p. We conclude that C^* is a completion time vector in $P|p_j = p|\sum_j w_j C_j$. □

A direct consequence of Theorem 8 is that the solutions in (3.4) give the following parallel-machine extension to Smith's WSPT rule for $P|p_j = p|\sum_j w_j C_j$.

Corollary 8. *An optimal solution for $P|p_j = p|\sum_j w_j C_j$ can be constructed by processing a job not yet processed with the largest weight whenever a machine is idle.*

For completeness, we construct a feasible solution to the dual of the linear program (3.1)-(3.2), and show that our dual solution obeys complementary slackness conditions with respect to the completion time vector given by Corollary 8. The dual LP is

$$\begin{aligned} \max \sum_{\mathcal{S} \subseteq \mathcal{N}} & f(\mathcal{S}) y_{\mathcal{S}} \\ \text{s.t.} \sum_{\mathcal{S} \subseteq \mathcal{N} : j \in \mathcal{S}} & y_{\mathcal{S}} = w_j \quad \forall j \in \mathcal{N} \\ & y_{\mathcal{S}} \geq 0 \quad \forall \mathcal{S} \subseteq \mathcal{N} \end{aligned}$$

with dual variables $y_{\mathcal{S}}$ for each primal constraint $\mathcal{S} \subseteq \mathcal{N}$. Let $n = |\mathcal{N}|$ and suppose without loss of generality that jobs are sorted in nonincreasing order of

w_j. The dual solution

$$\begin{aligned} y_{\{1\}} &= w_1 - w_2 \\ y_{\{1,2\}} &= w_2 - w_3 \\ &\vdots \\ y_{\{1,\ldots,n-1\}} &= w_{n-1} - w_n \\ y_{\{1,\ldots,n\}} &= w_n, \end{aligned}$$

with all other variables set to zero, is feasible. The corresponding primal constraints that hold with equality are

$$\sum_{j=1}^{k} C_j = f\left(\{1,\ldots,k\}\right) \quad \text{for each } k = 1,\ldots,n$$

which, as in our earlier discussion, implies that

$$\begin{aligned} C_j &= f\left(\{1,\ldots,j\}\right) - f\left(\{1,\ldots,j-1\}\right) \\ &= p\left(\left\lfloor \frac{j-1}{m} \right\rfloor + 1\right) \end{aligned}$$

for each job $j \in \mathcal{N}$.

Finally, we use the equivalence between an input for $P|p_j = p|\sum w_j C_j$ and a flipped input for $P||\sum C_j$, in which p_j and w_j are interchanged for every job j and the order in which jobs are processed is also reversed. This sorts jobs in *nondecreasing* order of *processing times*. SPT optimality follows directly from Theorem 8 and Corollary 8.

Corollary 9. *An optimal solution for $P||\sum C_j$ can be constructed by processing a job not yet processed with the shortest processing time whenever a machine is idle.*

3.3 Extensions

Our methods and the geometric insights therein may find further uses. We apply the same principle in two generalizations that are both well known to be polynomial-time solvable [17, 52, 53].

3.3.1 Uniform Machines

The first extension considers *uniform machines*, where each machine i has speed $s_i > 0$ and so processing job j on machine i takes p_j/s_i time units. This problem is denoted $Q||\sum C_j$. Much of the same principle applies in establishing an equivalence between $Q||\sum C_j$ and $Q|p_j = p|\sum w_j C_j$. A polyhedral approach similar to Theorem 8 can be used to solve the latter problem.

Observe that machines in $Q|p_j = p|\sum w_j C_j$ become idle at the following multiset of possible job completion times:

$$\left\{\frac{p}{s_1}, \ldots, \frac{p}{s_m}, \frac{2p}{s_1}, \ldots, \frac{2p}{s_m}, \ldots, \frac{np}{s_1}, \ldots, \frac{np}{s_m}\right\}.$$

Let $t_1 \leq t_2 \leq \cdots \leq t_n$ be the n smallest numbers in the above multiset. We use the following linear program for $Q|p_j = p|\sum w_j C_j$:

$$\begin{aligned} \min \ & \sum_{j \in \mathcal{N}} w_j C_j \\ \text{s.t.} \ & \sum_{j \in \mathcal{S}} C_j \geq \sum_{j=1}^{|\mathcal{S}|} t_j \quad \text{for all } \mathcal{S} \subseteq \mathcal{N}. \end{aligned} \tag{3.5}$$

The validity of the inequalities in (3.5) is immediate.

Theorem 9. *The valid inequalities in (3.5) define a supermodular polyhedron that completely describes $Q|p_j = p|\sum w_j C_j$.*

Proof. For ease of exposition, let $g(\mathcal{S}) = \sum_{j=1}^{|\mathcal{S}|} t_j$. We first show that g is supermodular. Observe that $g(\emptyset) = 0$. By definition, a function is supermodular if, $\forall A \subseteq B \subseteq \mathcal{N}$ and $i \notin B$,

$$g(B \cup \{i\}) - g(B) \geq g(A \cup \{i\}) - g(A).$$

By assumption, $|B| \geq |A|$, so

$$g(B \cup \{i\}) - g(B) = t_{|B|+1} \geq t_{|A|+1} = g(A \cup \{i\}) - g(A)$$

which establishes supermodularity.

Without loss of generality, suppose jobs are sorted in nonincreasing order of weights. The greedy algorithm for supermodular polyhedra implies that

$$\begin{aligned} C_j &= g\left(\{1, \ldots, j\}\right) - g\left(\{1, \ldots, j-1\}\right) \\ &= t_j \end{aligned}$$

for all $j \in \mathcal{N}$, so C is indeed a completion time vector in $Q|p_j = p| \sum w_j C_j$. □

By Theorem 9, an optimal schedule for $Q|p_j = p| \sum w_j C_j$ sorts jobs in nonincreasing order of weights and processes job k for completion at time t_k. When t_k takes the form $t_k = \ell p / s_i$, job k is the ℓth job scheduled on a machine with speed s_i. We therefore conclude that job k is the ℓth *last* job scheduled on machine i in an optimal schedule for $Q|| \sum C_j$.

3.3.2 Eligibility Constraints

Consider a generalization of $P|| \sum C_j$ in which each job j is compatible only with a subset of machines $\mathcal{M}_j$. We denote this problem $P|\mathcal{M}_j| \sum C_j$[1]. The

[1] This problem may also be denoted $R|p_{ij} \in \{p_j, \infty\}| \sum C_j$ as a special case of scheduling on *unrelated machines* where processing job j on machine i takes p_j units if $i \in \mathcal{M}_j$ and ∞ otherwise.

same principle from Section 3.1 can be used to establish an equivalence with $P|\mathcal{M}_j, p_j = p| \sum w_j C_j$.

We present a new result on SPT optimality given the following highly structured set of inputs where machine eligibility sets $\mathcal{M}_j$ are *nested*, and the highest-weight jobs are also the least restrictive, i.e., $w_1 \leq w_2 \leq \cdots \leq w_n$ and $|\mathcal{M}_1| \leq |\mathcal{M}_2| \leq \cdots \leq |\mathcal{M}_n|$ hold.

Theorem 10. *Suppose machine eligibility sets $\mathcal{M}_j$ are nested and jobs are sorted such that $w_1 \leq w_2 \leq \cdots \leq w_n$ and $|\mathcal{M}_1| \leq |\mathcal{M}_2| \leq \cdots \leq |\mathcal{M}_n|$ hold. For this highly structured set of inputs, an optimal solution for $P|\mathcal{M}_j, p_j = p| \sum w_j C_j$ can be constructed by inserting jobs over time, in sorted order, into the first slot in an eligible machine with the smallest sum of job weights.*

Proof. For a proof by contradiction, consider an optimal schedule that cannot be produced by this procedure. We show that we can always construct a schedule that follows this procedure that is as good as the optimal schedule.

Let W_{ij} denote the sum of job weights in machine i when job j is about to be scheduled. Let job j be the maximum-weight job in an optimal schedule that could not have been placed there in a schedule generated by the procedure. More precisely, we assume that job j is assigned to machine i when there exists some machine $\hat{\imath} \neq i$ such that $\hat{\imath} = \arg\min_{k \in \mathcal{M}_j} W_{kj}$. Let job $\hat{\jmath}$ be the job scheduled where job j should have been, that is, the first job scheduled in machine $\hat{\imath}$ after job j. If no such job $\hat{\jmath}$ exists, then job j must be the maximum-weight job in machine i: job j is the maximum-weight job that violates the procedure, and since $\mathcal{M}_j$ is nested, any job that comes after job j that is eligible for machine i is also eligible for machine $\hat{\imath}$. Finally, $W_{\hat{\imath}j} \leq W_{ij} \leq W_{ij} + w_j$, so a job must be scheduled in $\hat{\imath}$ before another can be scheduled in machine i. Reassigning job j

into the first slot of machine $\hat{\imath}$ places job j into a position compatible with the procedure and changes the objective by $(-W_{ij} + W_{\hat{\imath}j})p \leq 0$, which establishes a contradiction.

Suppose job $\hat{\jmath}$ exists, and let C_j and $C_{\hat{\jmath}}$ be the completion times of jobs j and $\hat{\jmath}$ in an optimal schedule, respectively. If $C_j \leq C_{\hat{\jmath}}$, swapping jobs j and $\hat{\jmath}$ changes the objective by

$$w_j C_{\hat{\jmath}} + w_{\hat{\jmath}} C_j - (w_j C_j + w_{\hat{\jmath}} C_{\hat{\jmath}}) = (w_j - w_{\hat{\jmath}})(C_{\hat{\jmath}} - C_j) \leq 0.$$

Otherwise, if $C_j > C_{\hat{\jmath}}$, we can swap the segment $[0, C_j)$ in machine i with the segment $[0, C_{\hat{\jmath}})$ in machine $\hat{\imath}$, which changes the objective by $(-W_{ij} + W_{\hat{\imath}j})(C_j - C_{\hat{\jmath}}) \leq 0$. In both cases, we place job j into a position compatible with the procedure and obtain a contradiction. Repeating this process for every job not compatible with the procedure gives an optimal solution. □

By the equivalence created by flipped inputs, an optimal solution for $P|\mathcal{M}_j|\sum C_j$ can be constructed by processing jobs in sorted order in an eligible machine with the shortest total processing time.

A primal-dual interpretation We conclude by outlining an LP-based approach for solving $P|\mathcal{M}_j, p_j = p|\sum w_j C_j$ for general inputs. This problem requires a new LP formulation that explicitly considers job-to-machine assignments. Define a binary variable x_{ijk} where $x_{ijk} = 1$ if job j is the kth job processed on machine i, and 0 otherwise. Let c_{ijk} denote the cost of this assignment such that $c_{ijk} = w_j kp$. Then the integer program for $P|\mathcal{M}_j, p_j = p|\sum_j w_j C_j$ is

$$
\begin{aligned}
\min \ & \sum_{j=1}^{n} \sum_{i \in \mathcal{M}_j} \sum_{k=1}^{n} c_{ijk} x_{ijk} \\
\text{s.t.} \ & \sum_{i \in \mathcal{M}_j} \sum_{k=1}^{n} x_{ijk} = 1 \qquad \forall j = 1, \ldots, n & (3.6) \\
& \sum_{j : i \in \mathcal{M}_j} x_{ijk} \leq 1 \qquad \forall i = 1, \ldots, m; \ k = 1 \ldots, n & (3.7) \\
& x_{ijk} \in \{0, 1\} \qquad \forall j = 1, \ldots, n; \ i \in \mathcal{M}_j; \ k = 1 \ldots, n.
\end{aligned}
$$

Constraint (3.6) ensures that every job is scheduled. By constraint (3.7), a machine can process at most one job at any given time. This is a bipartite matching problem with n jobs on one hand and nm machine-slot pairs on the other. It is well known that integrality constraints may be relaxed without altering the feasible region. The dual of the LP relaxation is

$$
\begin{aligned}
\max \ & \sum_{j=1}^{n} u_j - \sum_{i=1}^{m} \sum_{k=1}^{n} v_{ik} \\
\text{s.t.} \ & u_j \leq c_{ijk} + v_{ik} \qquad \forall j = 1, \ldots, n; \ i \in \mathcal{M}_j; \ k = 1 \ldots, n & (3.8) \\
& v_{ik} \geq 0 \qquad \forall i = 1, \ldots, m; \ k = 1 \ldots, n.
\end{aligned}
$$

Dual variables u_j and v_{ik} both have natural pricing interpretations: u_j is the total cost of assignment for job j, which includes both a baseline cost c_{ijk} and a premium v_{ik} attached to the k^{th} slot in machine i. Naturally, job j ultimately chooses an assignment that minimizes its total cost.

Primal-dual algorithms that solve minimum cost bipartite matching problems have been widely studied in the literature [68]. In what follows, we describe an iterative approach that led to the insights behind Theorem 10.

For each $j = 1, \ldots, n$, define a bipartite graph $G_j = (L_j, R, E_j)$ where $L_j =$

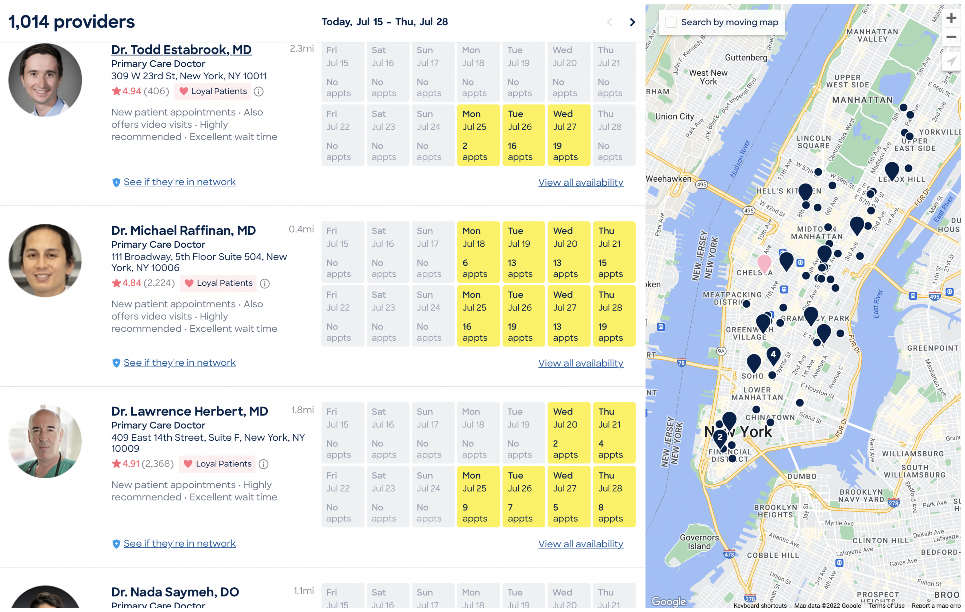
$\{1, \ldots, j\}$ is a subset of jobs, $R = \mathcal{M} \times \mathcal{N}$ is the set of machine-slot pairs, and $E_j = \{(\ell, ik) |\, \ell \in L_j, (i,k) \in R\}$. We initialize with an empty set of assignments $M = \emptyset$ and a dual feasible solution $u = v = 0$, and run a primal-dual matching algorithm on G_1. At each iteration $j = 2, \ldots, n$, we run the same algorithm on G_j with solutions obtained in the previous iteration as our initial feasible solutions. Upon termination, correctness follows automatically if $|M| = n$.

CHAPTER 4

THE VALUE OF FLEXIBILITY VIA JOIN THE SHORTEST-OF-D QUEUES

Imagine a patient who has an undiagnosed health concern who wants to see a physician. Oftentimes a patient's Primary Care Physician (PCP) is the first medical practitioner that she will contact to address her concern. Suppose that, unfortunately, the PCP is fully booked for the next couple of weeks and the patient will have to wait for a prolonged period of time to see the PCP. This long of a wait is not an uncommon scenario in the United States' healthcare system. According to a recent survey, the average wait time for a patient to see a doctor for non-emergency issues can be as long as 66 days in a large city [2]. Thus, the patient, in need of seeing a medical professional, might choose to forgo a visit with their PCP to see the next available physician and resolve their medical concern sooner. To this end, a patient may choose to use an online appointment scheduling platform such as ZocDoc where there is a large number of available doctors to choose from.

ZocDoc is a two-sided online medical platform that allows patients to search and view available appointment times of doctors and make appointments instantly. ZocDoc's sync technology [1] allows patients to search based on the doctor's location, medical specialty, insurance coverage, and patient ratings. On the ZocDoc patient platform, there are typically 10 doctors listed per page. In Figure 4.1, we provide an example of the ZocDoc platform where doctors are listed by earliest available appointment time and perceived quality. Appointment booking is not just online, but also can be made via smartphone devices as well. Doctors can also choose to be listed on ZocDoc and allow the platform to access and integrate with their appointment calendars so that their updated



1,014 providers

Today, Jul 15 – Thu, Jul 28

Dr. Todd Estabrook, MD 2.3mi
Primary Care Doctor
309 W 23rd St, New York, NY 10011
★4.94 (406) ♥ Loyal Patients ⓘ

New patient appointments · Also offers video visits · Highly recommended · Excellent wait time

Fri Jul 15 No appts	Sat Jul 16 No appts	Sun Jul 17 No appts	Mon Jul 18 No appts	Tue Jul 19 No appts	Wed Jul 20 No appts	Thu Jul 21 No appts
Fri Jul 22 No appts	Sat Jul 23 No appts	Sun Jul 24 No appts	Mon Jul 25 2 appts	Tue Jul 26 16 appts	Wed Jul 27 19 appts	Thu Jul 28 No appts

See if they're in network

View all availability

Dr. Michael Raffinan, MD 0.4mi
Primary Care Doctor
111 Broadway, 5th Floor Suite 504, New York, NY 10006
★4.84 (2,224) ♥ Loyal Patients ⓘ

New patient appointments · Also offers video visits · Highly recommended · Excellent wait time

Fri Jul 15 No appts	Sat Jul 16 No appts	Sun Jul 17 No appts	Mon Jul 18 6 appts	Tue Jul 19 13 appts	Wed Jul 20 13 appts	Thu Jul 21 15 appts
Fri Jul 22 No appts	Sat Jul 23 No appts	Sun Jul 24 No appts	Mon Jul 25 16 appts	Tue Jul 26 19 appts	Wed Jul 27 13 appts	Thu Jul 28 19 appts

See if they're in network

View all availability

Dr. Lawrence Herbert, MD 1.8mi
Primary Care Doctor
409 East 14th Street, Suite F, New York, NY 10009
★4.91 (2,368) ♥ Loyal Patients ⓘ

New patient appointments · Highly recommended · Excellent wait time

Fri Jul 15 No appts	Sat Jul 16 No appts	Sun Jul 17 No appts	Mon Jul 18 No appts	Tue Jul 19 No appts	Wed Jul 20 2 appts	Thu Jul 21 4 appts
Fri Jul 22 No appts	Sat Jul 23 No appts	Sun Jul 24 No appts	Mon Jul 25 9 appts	Tue Jul 26 7 appts	Wed Jul 27 5 appts	Thu Jul 28 8 appts

See if they're in network

View all availability

Dr. Nada Saymeh, DO 1.1mi
Primary Care Doctor

Fri Jul 15	Sat Jul 16	Sun Jul 17	Mon Jul 18	Tue Jul 19	Wed Jul 20	Thu Jul 21

Figure 4.1: A snapshot of the ZocDoc platform

calendars can be viewed by patients in real-time. From a patient's perspective, using a service like ZocDoc can help patients book appointments sooner. Earlier appointments typically result in earlier detection of illnesses, which can affect the final cost of healthcare expenses. Thus, we ask the question, what is the value of being able to see another physician on a patient platform like ZocDoc if one is flexible?

In this chapter, we abstract the above scenario and model it as a multi-server queueing system under heavy traffic and partial load balancing. Similar types of queueing systems have been studied in the literature, see for example Whitt [85], Vvedenskaya et al. [84], Lin and Raghavendra [54], Mitzenmacher [59], Graham [37, 36, 38], Foley and McDonald [32], Mitzenmacher [60], Graham [39], Dai et al. [21], He and Down [43], Bramson [14], Lu et al. [56], Tsitsiklis and Xu [81], Bramson et al. [15], Mukherjee et al. [64], Aghajani et al. [3], Tao and Pender

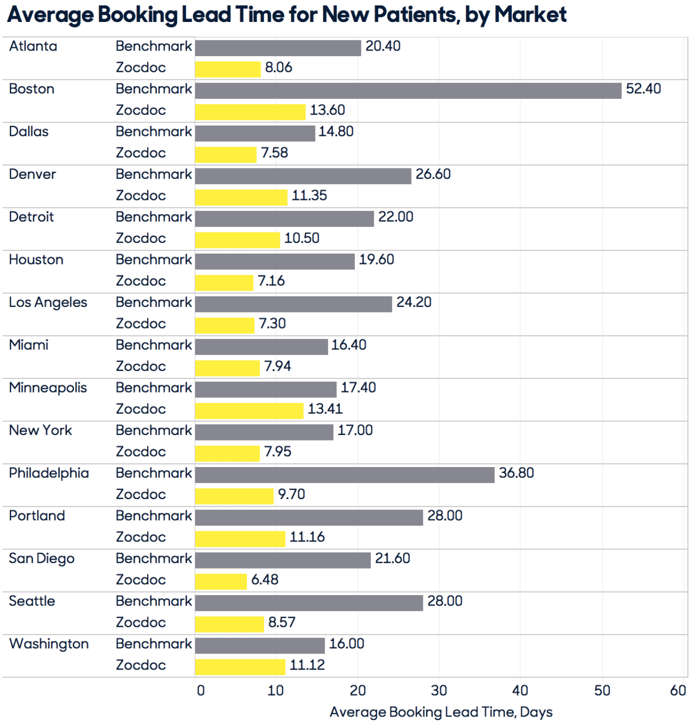


Figure 4.2: ZocDoc's success in reducing wait lead times

[79], Foss and Stolyar [33].

However, we analyze the performance of the multi-server queue with the addition of a key feature: patient types, where patients of different types react differently to the idea of waiting to see their PCP. For example, some patients are quite particular about only being seen by their PCP for various reasons. These reasons might include familiarity, ease of communication, and accessibility of location. On the other hand, there also exist *flexible* patients who are willing to see another physician other than their PCP if they can have access to a medical professional within a shorter time frame. In fact, these flexible patients might be willing to call several physician's offices, observe waiting times for each physician, and finally *join the queue* by scheduling an appointment with the physician that offers the shortest wait time among those contacted or listed on the ZocDoc platform. In the context of ZocDoc, this flexibility decreases the booking lead time significantly for those who are willing to be flexible [1]. In Figure 4.2, we observe that the rising popularity of ZocDoc as a patient platform has reduced the wait lead time significantly by offering substitute doctors who are willing to see the patient in an earlier time frame. In doing so, flexible patients will acquire queue length information that dedicated patients do not have access to. The question we aim to address in this chapter is, how does the overall system perform if only a fraction $p \in [0, 1]$ of the patients are flexible and are willing to use a platform like ZocDoc?

The model that we consider is highly stylized. We consider a system of N physicians and assume that patients who arrive to the system are one of two types: flexible or dedicated. We fix a flexibility parameter $p \in [0, 1]$, which denotes the probability with which each arriving patient is flexible. We further assume that each patient type has a different policy for joining a physician's queue. The dedicated patients join their designated PCP's queue regardless of queue length, i.e., they join one of the N queues uniformly at random. In other words, these patients either have **no information** and do not use a platform like ZocDoc to search for earlier appointments. They are in some sense loyal to their PCP regardless of the wait they might experience. Flexible patients, on the other hand, are willing to see any physician that reduces their waiting time and are considered impatient. In our model, flexible patients choose d physicians, independently and uniformly at random, and observe the queue lengths of each. Flexible patients subsequently respond to this newly obtained information by joining the shortest queue among the d physician queues sampled. In some of the current literature, the parameter d scales with the number of servers N, see for example Dieker and Suk [25]. However, we assume that d is a fixed constant since the ZocDoc patient platform displays roughly 10 physicians at a time on each page. The value $d = 10$ is therefore a reasonable value for the purposes of our work.

Our goal is to study the performance of the system for varying degrees of *flexibility* and *power of choices*, as expressed by parameters p and d, respectively. In doing so, we use a fluid approximation where the queue length dynamics are approximated with a deterministic fluid model as $N \to \infty$ and the fluid model behaves according to an infinite dimensional system of non-linear ordinary differential equations. We are especially interested in studying and deriving an upper bound for the average queue length in the system, which, as we will see, also has some interesting interpretations.

In addition to the healthcare motivation presented by the ZocDoc platform, one can also imagine a supermarket where customers join lines independently without any knowledge of the number of customers at each cashier. Our model is equivalent to having a proportion of informed customers who have the ability to look at d queues and join the shortest among those queues. Thus, our goal is to understand the value that a few *informed* customers can have on the system. We will show in the sequel that even when the proportion of flexible patients is small, these flexible patients can have a large impact on the overall system performance.

Related Work There has been a lot of activity in the recent years of researchers analyzing a number of variants of the join the shortest queue model. See, for example, recent work by Eschenfeldt and Gamarnik [30], Braverman [16], Mukherjee et al. [65], Banerjee and Mukherjee [11]. Despite the large amount of activity in this area, there are relatively few papers that explore the impact of flexibility or information in the underlying system. This chapter is inspired by the work of Tsitsiklis and Xu [81] where they explicitly study the trade-off between centralized and distributed processing. In their work, they consider an N-station system where their system designer is given a total amount N of divisible computing resources. Moreover, the system designer in their work can allocate resources to local and central servers. More specifically, for some fraction $p \in (0, 1)$, local servers process tasks at a maximum rate of $1-p$ tasks per second, while the centralized server processes tasks at rate of pN tasks per second. Our work is different from theirs in two main ways. First, we consider a different model where we are joining the shortest of d queues. Second, we do not assume a centralized server processes tasks. In our setting, **flexibility** can be viewed as information each arrival has about the system. Some customers have some partial information about the system and the others do not have any information about the system and join uniformly at random. We also differ from Tsitsiklis and Xu [81] since we also analyze the diffusion scaled system. By studying the diffusion scaled process, we are able to gain important insights on how flexibility impacts the fluctuations or variance of the queueing system. This is also helpful in building confidence intervals around the fluid limit.

Main Contributions The main contributions in this work are:

- We develop a new stochastic queueing model that incorporates the structure of dedicated and flexible customers. We explore the trade-off between these types of customers through the parameters p and d, which represent flexibility and the amount of partial information about the system.
- We prove fluid and diffusion limit theorems for the queueing process,

thus showing that the fluid limit is an infinite dimensional system of non-linear odes and that the diffusion limit is an infinite dimensional Ornstein-Uhlenbeck process.

- We prove an interchange of limits for the fluid and diffusion scaled processes, thereby showing that the steady state fluid and diffusion limits are good approximations for the original fluid and diffusion scaled processes. In fact, we derive a closed form expression for the steady state distribution using a non-linear recursion. This recursion also allows us to derive new upper and lower bounds on the first and second moments of the queue length in steady state, which converge to each other as $p \to 0$ or $p \to 1$.
- From a mathematical perspective, we derive a new method for proving the global stability of the steady state fluid limit by using a comparison approach. Our approach exploits the fact that if the integral of the difference of two solutions are bounded, then the two solutions converge to the same point. We also derive new infinite horizon bounds for the diffusion scaled process, which are important ingredients for establishing tightness for steady state diffusion limits. The infinite horizon bounds are in general difficult to prove because they must be proved in the appropriate functional space when the sub-generator of an associated birth-death process is not self-adjoint. Moreover, proving these infinite horizon bounds is difficult in our model because the self-adjoint property of the sub-generator depends on the flexibility parameter p.

The remainder of this chapter is as follows. In Section 4.1, we describe the stochastic model. In Section 4.2, we present a fluid model for the tail distribution of the queue length. We prove both a transient and a steady state fluid limit for our stochastic model. The transient fluid limit is proved using martingale techniques and the steady state fluid limit is proved using a new comparison approach. We also prove an interchange of limits, which shows in a rigorous sense that the steady state limit can be used as an approximation for our stochastic model. In Section 4.3, we present a diffusion model for the tail distribution of the queue length and prove a transient diffusion limit, a steady state diffusion limit, and an interchange of limits for the stochastic model. In Section 4.4, we prove that the steady state fluid limit can be written in closed form using a nonlinear recursion. We also prove tight upper and lower bounds on the first and second moments of the queue length. We also demonstrate through numerical examples that small values of p can have a large impact on the behavior the system. Section 4.5 concludes.

Notation The following table summarizes the notations that will be used throughout this chapter.

Table 4.1: Notation

N	# of physicians
λ	Arrival rate of patients
p	Fraction of flexible patients
d	# of physicians flexible patients sample
$Q_i^N(t)$	Number of patients at physician i at time t
$S_i^N(t)$	Fraction of queues with at least i patients at time t
$s_i(t)$	The fluid limit of process $S_i^N(t)$
s^I	The steady state of fluid limit $s(t)$
$D^N(t)$	The fluctuation of $S^N(t)$ around its fluid limit $s(t)$
$D(t)$	The diffusion limit of process $D^N(t)$
ℓ_1	The space of sequences whose series is absolutely convergent
ℓ_2	The space of square-summable sequences
$\mathcal{S}$	$\{s \in [0,1]^{\mathbb{Z}_+} : 1 \geq s_0 \geq s_1 \geq \cdots \geq 0, \sum_{i=0}^{\infty} s_i < \infty\}$

Preliminaries of Weak Convergence In this chapter, we assume that all random variables are defined on a common probability space $(\Omega, \mathcal{F}, \mathbb{P})$. Moreover, for all positive integers k, we let $\mathcal{D}([0,\infty),\mathcal{S})$ be the space of right continuous functions with left limits (RCLL) in $\mathcal{S}$ that have a time domain in $[0,\infty)$. As is usual, we endow the space $\mathcal{D}([0,\infty),\mathcal{S})$ with the usual Skorokhod J_1 topology, and let M be defined as the Borel σ-algebra associated with the J_1 topology. We also assume that all stochastic processes are measurable functions from our common probability space $(\Omega,\mathcal{F},\mathbb{P})$ into $(\mathcal{D}([0,\infty),\mathcal{S}),M)$. Thus, if $\{\zeta\}_{n=1}^{\infty}$ is a sequence of stochastic processes, then the notation $\zeta^n \to \zeta$ implies that the probability measures that are induced by the ζ^n's on the space $(\mathcal{D}([0,\infty),\mathcal{S}),M)$ converge weakly to the probability measure on the space $(\mathcal{D}([0,\infty),\mathcal{S}),M)$ induced by ζ. For any $x \in (\mathcal{D}([0,\infty),\mathcal{S}),M)$ and any $T > 0$, we define

$$||x||_{\ell_2} \equiv \sum_{i=0}^{\infty} x_i^2 \tag{4.1}$$

and note that ζ^n converges almost surely to a continuous limit process ζ in the J_1 topology if and only if

$$||\zeta^n - \zeta||_{\ell_2} \to 0 \quad a.s. \tag{4.2}$$

4.1 A Stochastic Queueing Model

In this section, we present a stochastic queueing model that has N physicians. Each physician operates a single server queue of scheduled patients who are seen in a first in first out manner. We denote the queue length for physician n at time t with $Q_n(t)$ where $n \in \{1,2,\cdots,N\}$ and $t \geq 0$. Each physician processes the work of their current patients at rate 1 if there are patients in their queue.

For the patients, we assume there are two types of patients: *dedicated* and *flexible*. The two types of patients are split into according to our flexibility parameter p. A patient is flexible with fixed probability $p \in [0,1]$. We assume that flexible patients are willing to sample d physician queues, independently and uniformly at random, and join the shortest-of-d queues at their time of arrival. This is an abstraction of patients choosing among the available physicians on the ZocDoc platform. Dedicated patients, on the other hand, are only willing to see their designated PCP and are not flexible. Thus, assuming equal popularity among all physicians, this is equivalent to saying that they join any queue at random. Finally, we assume that once a patient joins a queue, the patient is completely locked in and cannot switch to another queue.

Each of the N physicians has a stream of dedicated patients arriving according to independent Poisson processes with a common rate $\lambda(1-p)$, where $\lambda \in [0,1]$. Thus, the total arrival rate of dedicated patients to the system is $\lambda(1-p)N$. In addition, the overall system also has a stream of flexible patients arriving according to an independent Poisson process with rate $\lambda p N$.

Once patients are routed to the appropriate physician queue (dedicated patients to their PCP queues, and flexible patients to the shortest-of-d physician queues), each physician queue operates as an $M/M/1$ queue. The queue length vector at time t, $(Q_1(t), Q_2(t), \cdots, Q_N(t))$, is a Markov process. In addition, the system is fully symmetric and exchangeable in that the arrival of dedicated patients and patient services are independent and identical, and the arrival of flexible patients depends solely on the length of the physician queues, and not on the specific identity of physicians. Thus, we can use a Markov process $\{S_i^N(t)\}_{i=0}^{\infty}$ to describe the evolution of the system, where we defined $S_i^N(t)$ as

$$S_i^N(t) = \frac{1}{N}\sum_{n=1}^{N} \mathbf{1}\{Q_n(t) \geq i\}. \tag{4.3}$$

Here $S_i^N(t)$ represents the fraction of queues with at least i patients. By definition, $S_0^N(t) = 1$ for all values of N and $t \geq 0$. Furthermore, $S_i^N(t)$ is a nonincreasing process in the variable i, meaning that

$$1 \geq S_i^N(t) \geq S_{i+1}^N(t) \geq 0$$

for all values of i, N and $t \geq 0$. We define the infinite dimensional vector of this queueing process as $S^N(t) = (S_0^N(t), S_1^N(t), \cdots, S_n^N(t), \cdots S_\infty^N(t))$. Our goal is to study the process $S^N(t)$ in two scenarios. The first is in the transient setting where we let $N \to \infty$ and the second is in the steady state setting where we let both $N \to \infty$ and $t \to \infty$.

4.2 Fluid Model

Here we summarize the results in this section, which are related to the fluid model of the queueing process $S^N(t)$. In Theorem 11, we prove a functional law of large numbers (LLN) in the transient case for process $S^N(t)$ to its fluid limit $s(t)$. In Theorem 14, we prove an interchange of limits for the stochastic process model. We use a compactness-uniqueness approach, which shows that the limiting point s^I of the fluid limit $s(t)$ is also the limit of the invariant measure $S^N(\infty)$ of $S^N(t)$.

4.2.1 Transient Analysis

We start with the functional law of large numbers in the transient case for the fluid limit.

Theorem 11 (Functional Law of Large Numbers). *Assume that $(S^N(0))_{N\geq d}$ converges in distribution to $s(0)$ in $\mathcal{S}$. Then, $(S^N(t))_{N\geq d}$ converges in probability to the* *unique solution $s = (s(t))_{t\geq 0}$ i.e. on any compact time interval $t_0 > 0$ and $\epsilon > 0$, we have*

$$\lim_{N\to\infty} \mathbb{P}\left(\sup_{t\leq t_0} \|S^N(t) - s(t)\|_{\ell_2} > \epsilon\right) = 0. \tag{4.4}$$

Moreover, $s(t)$ has initial condition $s(0)$ and is the solution to the following infinite dimensional system of differential equations

$$\frac{ds_i}{dt} = \underbrace{\lambda(1-p)(s_{i-1} - s_i)}_{\textit{arrival of dedicated patients}} + \underbrace{\lambda p\left(s^d_{i-1} - s^d_i\right)}_{\textit{arrival of flexible patients}} - \underbrace{(s_i - s_{i+1})}_{\textit{departure of patients}} \qquad i \geq 1. \tag{4.5}$$

Proof. We prove this result using Doob's inequality for martingales and Gronwall's lemma. We use Proposition 4 and Lemma 8 in the proof, which are stated after the proof of Theorem 11. To give readers a high-level understanding of the proof idea, we list the essential steps and the related theorem numbers below.

1. We decompose the queueing process $S^N(t)$ into three parts. The first is the initial condition $S^N(0)$, the second is a martingale $M^N(t)$ term, and the final term is an integral of the drift term $\int_0^t F^N(S^N(u))du$. (Equation 4.6)
2. We bound the difference between $S^N(t)$ and its fluid limit $s(t)$ on any finite interval $[0, T]$ by the difference in their initial conditions $\|S^N(0) - s(0)\|$, the supremum of martingale $\sup_{u\leq T} \|M^N(t)\|$, the difference in drift function and its limit $\int_0^T \|F^N(S^N(u)) - F(S^N(u))\|du$, and finally the difference in limiting drift function evaluated at $S^N(t)$ and $s(t)$, i.e., $\int_0^T \|F(S^N(u)) - F(s(u))\|du$. (Inequality 4.7)
3. We show that the limiting drift function F is Lipschitz. (Proposition 4)
4. We apply Gronwall's lemma to the difference. (Inequality 4.8)
5. We apply Doob's L_2 martingale inequality to $M^N(t)$ and the bounds on quadratic variation. (Inequality 4.10, Lemma 8)

6. We prove existence and uniqueness of the fluid limit $s(t)$. (Proposition 5)

We begin by introducing the falling factorial notation $(x)_k = x(x-1)\cdots(x-k+1)$ for $x \in \mathbb{R}$, and define the following mappings for s in c_0:

$$\begin{aligned}
F_+(s)(i) &= \lambda(1-p)(s_{i-1} - s_i) + \lambda p(s_{i-1}^d - s_i^d), \\
F_-(s)(i) &= (s_i - s_{i+1}), \\
F_+^N(s)(i) &= \lambda(1-p)(s_{i-1} - s_i) + \lambda p \frac{(Ns_{i-1})_d - (Ns_i)_d}{(N)_d}, \quad i \geq 1, \\
F^N(s) &= F_+^N(s) - F_-(s), \\
F(s) &= F_+ s) - F_-(s).
\end{aligned}$$

Then, the nonlinear differential equation can be written as $\dot{s} = F(s)$. It is easy to show that $S^N(t)$ is a Markov process that, when in state s, has a jump in the i^{th} coordinate of size $+1/N$ with rate $NF_+^N(s)(i)$ and of size $-1/N$ with rate $NF_-(s)(i)$, for all $i \geq 1$. Since $S^N(t)$ is a semi-martingale, we have the following decomposition of $S^N(t)$,

$$S^N(t) = \underbrace{S^N(0)}_{\text{initial condition}} + \underbrace{M^N(t)}_{\text{martingale}} + \int_0^t \underbrace{F^N(S^N(u))}_{\text{drift term}} du, \tag{4.6}$$

where $S^N(0)$ is the initial condition and $M^N(t)$ is a independent family of martingales. Moreover, $\int_0^t F^N(S^N(u))du$ is the integral of the drift term where the drift term is given by $F^N : \mathcal{S} \to \mathbb{R}^{\mathbb{Z}_+}$ or

$$\begin{aligned}
F^N(s)(k) &= \sum_{x \neq s} (x - s) Q^N(s, x)(k) \\
&= \lambda(1-p)(s_{k-1} - s_k) + \lambda p \frac{(Ns_{k-1})_d - (Ns_k)_d}{(N)_d} - (s_k - s_{k+1}),
\end{aligned}$$

where $Q^N(s, x)(k)$ represents the transition rate from state s to x on the k^{th} coordinate. Now we want to compare $S^N(t)$ with its fluid limit $s(t)$ defined by

$$s(t) = s(0) + \int_0^t F(s(u))du.$$

Letting $\|\cdot\|$ denote the ℓ_2 norm in $\mathbb{R}^{\mathbb{Z}_+}$,

$$\begin{aligned}\left\|S^N(t)-s(t)\right\| &= \left\|S^N(0)+M^N(t)+\int_0^t F^N(S^N(u))du - s(0) - \int_0^t F(s(u))du\right\| \\ &= \left\|S^N(0)-s(0)+M^N(t)+\int_0^t \left(F^N(S^N(u))-F(S^N(u))\right)du \right. \\ &\quad \left. +\int_0^t (F(S^N(u))-F(s(u)))du\right\|.\end{aligned}$$

Now we define the random function $f^N(t) = \sup_{u\le t}\left\|S^N(u)-s(u)\right\|$. By the triangle inequality we have

$$\begin{aligned}f^N(t) &\le \|S^N(0)-s(0)\| + \sup_{u\le t}\|M^N(u)\| + \int_0^t \|F^N(S^N(u))-F(S^N(u))\|du \\ &\quad +\int_0^t \|F(S^N(u))-F(s(u))\|du. \end{aligned} \tag{4.7}$$

By Proposition 4, $F(s)$ is Lipschitz with respect to ℓ_2 norm. Let L be the Lipschitz constant of $F(s)$, then

$$\begin{aligned}f^N(t) &\le \|S^N(0)-s(0)\| + \sup_{u\le t}\|M^N(u)\| + \int_0^t \|F^N(S^N(u))-F(S^N(u))\|du \\ &\quad +\int_0^t \|F(S^N(u))-F(s(u))\|du \\ &\le \|S^N(0)-s(0)\| + \sup_{u\le t}\|M^N(u)\| + \int_0^t \|F^N(S^N(u))-F(S^N(u))\|du \\ &\quad +L\int_0^t \|S^N(u)-s(u)\|du \\ &\le \|S^N(0)-s(0)\| + \sup_{u\le t}\|M^N(u)\| + \int_0^t \|F^N(S^N(u))-F(S^N(u))\|du \\ &\quad +L\int_0^t f^N(u)du.\end{aligned}$$

By Gronwall's lemma,

$$f^N(t) \le \left(\|S^N(0)-s(0)\| + \sup_{u\le t}\|M^N(u)\| + \int_0^t \|F^N(S^N(u))-F(S^N(u))\|du\right)e^{Lt}. \tag{4.8}$$

Now we proceed to bound $f^N(t)$ term by term. To this end, we define function

$\alpha : \mathcal{S} \to \mathbb{R}^{\mathbb{Z}_+}$ as

$$\begin{aligned}
\alpha(s)(k) &= \sum_{x \neq s} \|x-s\|^2 Q^N(s,x)(k) \\
&= \frac{1}{N}\left[F_+^N(s)(k) + F_-(s)(k)\right] \\
&= \frac{1}{N}\left[\lambda(1-p)(s_{k-1}-s_k) + \lambda p \frac{(Ns_{k-1})_d - (Ns_k)_d}{(N)_d} + (s_k - s_{k+1})\right].
\end{aligned}$$

By Lemma 8, we have that $\|\alpha(s)\|_{\ell_2} = \frac{1}{N}O(\|s\|_{\ell_2})$. Thus, there exist a constant $C > 0$ such that $\|\alpha(s)\|_{\ell_2} \leq \frac{C}{N}$ for any s. Now consider the following four sets

$$\begin{aligned}
\Omega_0 &= \{\|S^N(0) - s(0)\| \leq \delta\}, \\
\Omega_1 &= \left\{\int_0^{t_0} \|F^N(S^N(t)) - F(S^N(t))\| dt \leq \delta\right\}, \\
\Omega_2 &= \left\{\int_0^{t_0} \|\alpha(S^N(t))\| dt \leq A(N)t_0\right\}, \\
\Omega_3 &= \left\{\sup_{t \leq t_0} \|M^N(t)\| \leq \delta\right\},
\end{aligned}$$

where $\delta = \epsilon e^{-Lt_0}/3$. Here, the set Ω_0 is for bounding the initial condition, the set Ω_1 is for bounding the drift term F^N and the limit of the drift term F, and the sets Ω_2 and Ω_3 are for bounding the martingale $M^N(t)$. Therefore, on the event $\Omega_0 \cap \Omega_1 \cap \Omega_3$,

$$f^N(t_0) \leq 3\delta e^{Lt_0} = \epsilon. \tag{4.9}$$

Consider the stopping time

$$T = t_0 \wedge \inf\left\{t \geq 0 : \int_0^t \alpha(S^N(u))du > A(N)t_0\right\},$$

by Doob's ℓ_2 martingale inequality,

$$\mathbb{E}\left(\sup_{t \leq T} \|M^N(t)\|^2\right) \leq 4\mathbb{E}\|M^N(T)\|^2 = 4\int_0^T \|\alpha(S^N(u))\| du. \tag{4.10}$$

On Ω_2, we have $T = t_0$, so $\Omega_2 \cap \Omega_3^c \subset \{\sup_{t \leq T} \|M^N(t)\| > \delta\}$. By Chebyshev's inequality,

$$\mathbb{P}(\Omega_2 \cap \Omega_3^c) \leq \mathbb{P}\left(\sup_{t \leq T} \|M^N(t)\| > \delta\right) \leq \frac{\mathbb{E}\left(\sup_{t \leq T} \|M^N(t)\|^2\right)}{\delta^2} \leq 4A(N)t_0/\delta^2.$$

Thus, by Equation (4.9), we have the following result,

$$\begin{aligned}\mathbb{P}\left(\sup_{t\le t_0}\|S^N(t)-s(t)\|>\epsilon\right) &\le \mathbb{P}(\Omega_0^c\cup\Omega_1^c\cup\Omega_3^c)\\ &\le \mathbb{P}(\Omega_2\cap\Omega_3^c)+\mathbb{P}(\Omega_0^c\cup\Omega_1^c\cup\Omega_2^c)\\ &\le 4A(N)t_0/\delta^2+\mathbb{P}(\Omega_0^c\cup\Omega_1^c\cup\Omega_2^c)\\ &= 36A(N)t_0e^{2Lt_0}/\epsilon^2+\mathbb{P}(\Omega_0^c\cup\Omega_1^c\cup\Omega_2^c).\end{aligned}$$

Let $A(N)=\frac{C}{N}$, then $\Omega_2^c=\emptyset$. And since $S^N(0)\xrightarrow{p}s(0)$, $\lim_{N\to\infty}\mathbb{P}(\Omega_2^c)=0$. Therefore we have

$$\lim_{N\to\infty}\mathbb{P}\left(\sup_{t\le t_0}\|S^N(t)-s(t)\|>\epsilon\right)=\lim_{N\to\infty}\mathbb{P}(\Omega_1^c).$$

By Lemma 8, $\lim_{N\to\infty}\mathbb{P}(\Omega_1^c)=0$, so

$$\lim_{N\to\infty}\mathbb{P}\left(\sup_{t\le t_0}\|S^N(t)-s(t)\|>\epsilon\right)=0.$$

□

Proposition 4 (Lipschitz bound on drift functions)**.** *The mappings F, F_+, F_- are Lipschitz with respect to the ℓ_2 norm.*

Proof. By the identity $u^d-v^d=(u-v)(u^{d-1}+u^{d-2}v+\cdots+v^{d-1})\le d(u-v)$, we have the Lipschitz bound

$$\begin{aligned}\|F_+(u)-F_+(v)\|_{\ell_2}^2 &\le \sum_{i=1}^{\infty}\big(\|\lambda(1-p)(u_{i-1}-v_{i-1})\|^2+\|\lambda(1-p)(u_i-v_i)\|^2\\ &\quad+\|\lambda p(u_{i-1}^d-v_{i-1}^d)\|^2+\|\lambda p(u_i^d-v_i^d)\|^2\big)\\ &\le 4\sum_{i=0}^{\infty}\left[\lambda^2(1-p)^2\|u_i-v_i\|^2+(\lambda pd)^2\|u_i-v_i\|^2\right]\\ &\le 8\lambda^2d^2\|u-v\|_{\ell_2}^2.\end{aligned}$$

Similarly we can show that

$$\|F_-(u)-F_-(v)\|_{\ell_2}^2 \le 2\|u-v\|_{\ell_2}^2.$$

Thus, the mappings F, F_+, F_- are Lipschitz with respect to the ℓ_2 norm. □

Proposition 5 (Existence and Uniqueness of the fluid limit)**.** *There exists a unique solution $(s(t))_{t\geq 0} \in \mathcal{S}$ to the differential equation (4.5) and $s(t)$ is continuous in t.*

Proof. This is a direct application from the Lipschitz property of F in Proposition 4 and Gronwall's lemma. □

4.2.2 Steady State Analysis

In addition to understanding the transient behavior of the fluid model, it is important to understand the steady state behavior as well. In this section, we outline the steady state analysis of the stochastic queueing model. We first denote the steady state of the queueing model as s^I. Then, s^I satisfies the following equation,

$$\lambda(1-p)(s^I_{i-1} - s^I_i) + \lambda p\left((s^I_{i-1})^d - (s^I_i)^d\right) - (s^I_i - s^I_{i+1}) = 0, \quad i \geq 1 \tag{4.11}$$

Theorem 12. *The steady state of the queueing model s^I has a unique solution given by the following recursion*

$$s^I_0 = 1,$$

$$s^I_1 = \lambda,$$

$$s^I_i = \lambda(1-p)s^I_{i-1} + \lambda p(s^I_{i-1})^d \quad \textit{for all } i \geq 2.$$

Proof. We prove the result by induction. For $i = 1$,

$$\begin{aligned} s_1^I &= \sum_{i=1}^{\infty}(s_i^I - s_{i+1}^I) \\ &= \sum_{i=1}^{\infty}\left[\lambda(1-p)(s_{i-1}^I - s_i^I) + \lambda p\left((s_{i-1}^I)^d - (s_i^I)^d\right)\right] \\ &= \lambda(1-p)s_0^I + \lambda p(s_0^I)^d \\ &= \lambda. \end{aligned}$$

Now for $i \leq k$, we assume that

$$s_i^I = \lambda(1-p)s_{i-1}^I + \lambda p(s_{i-1}^I)^d.$$

Then, for $i = k+1$,

$$\begin{aligned} s_{k+1}^I &= s_k^I - \lambda(1-p)(s_{k-1}^I - s_k^I) - \lambda p\left((s_{k-1}^I)^d - (s_k^I)^d\right) \\ &= \lambda(1-p)s_k^I + \lambda p(s_k^I)^d. \end{aligned}$$

□

Remark: Note that the existence and uniqueness of the equilibrium point s^I is obtained from the fact that s_i^I is completely determined by s_{i-1}^I and we have the initial condition $s_0^I = 1$ holds.

4.2.3 Interchanging Limits

In this section, we prove an interchange of limits for the fluid model, i.e. the limiting point s^I of the fluid limit $s(t)$ is also the limit of the invariant measure $S^N(\infty)$ of $S^N(t)$. A visual interpretation of the interchange of limits result corresponds to showing that the following diagram commutes.

$$\begin{tikzcd}
S^N(t) \arrow[r, "N\to\infty"] \arrow[d, "t\to\infty"'] & s(t) \arrow[d, "t\to\infty"] \\
S^N(\infty) \arrow[r, "N\to\infty"'] & s^I
\end{tikzcd}$$

We have already proved in Section 4.2 that $S^N(t) \xrightarrow{p} s(t)$ and the existence and uniqueness of s^I. Now we will show the other two directions of the diagram, which are the existence of invariant measure $S^N(\infty)$ for each $N \geq 1$, and the convergence of the invariant measure $S^N(\infty)$ to s^I. Our method of proof is a modification of the compactness-uniqueness method pioneered by Graham [36]. We can decompose the compactness uniqueness method into three essential steps.

1. Show that the fluid limit (Equation (4.5)) has a globally attractive stable point s^I. (Lemma 5, Theorem 13)
2. Show that there exists an invariant measure $S^N(\infty)$ for S^N for each $N \geq 1$. (Proposition 6, Theorem 14 (1))
3. Show that these invariant measures $(S^N(\infty))_{N\geq 1}$ are tight in $\mathcal{S}$. (Theorem 14 (2))

In order to prove that the fluid limit has a globally attractive stable point, we will use a comparison result for finite dimensional ordinary differential equations. This result is outlined below.

Lemma 5 (Comparison Result)**.** *Let u and v be two solutions for Equation (4.5) such that $u(0) \leq v(0)$. Then $u(t) \leq v(t)$ for all $t \geq 0$.*

Proof. We first consider the finite dimensional case. For any fixed constant $K \in \mathbb{N}$, we assume WLOG that $u_k(0) < v_k(0), k = 1, \cdots, K$, and that $u_{K+1}(t) < v_{K+1}(t)$ for all $t \geq 0$. We aim to show that $u_k(t) < v_k(t)$ for all $t \geq 0$ and $k = 1, \cdots, K$.

Assume that $u(t) < v(t)$ for $t \in [0, t_0)$ but $u_i(t_0) = v_i(t_0)$ for some $i \in \{1, \cdots, K\}$. Then we know that $u_j(t_0) \leq v_j(t_0)$ for all $j \in \{1, \cdots, K\}$. Now from the fluid limit equation (4.5) we have that

$$\begin{aligned}\dot{u}_i(t_0) &= \lambda(1-p)(u_{i-1}(t_0) - u_i(t_0)) + \lambda p(u_{i-1}^d(t_0) - u_i^d(t_0)) - (u_i(t_0) - u_{i+1}(t_0)) \\ &\leq \lambda(1-p)(v_{i-1}(t_0) - v_i(t_0)) + \lambda p(v_{i-1}^d(t_0) - v_i^d(t_0)) - (v_i(t_0) - v_{i+1}(t_0)) \\ &= \dot{v}_i(t_0),\end{aligned}$$

suggesting that $u_i(t) \leq v_i(t)$ for $t \geq t_0$.

Now for any $s(0) \in \mathcal{S}$, there exists a unique solution $s(t) \in \mathcal{S}$ for (4.5). We will show that the solution $s(t)$ can be obtained as the limit of solutions $\{s^K(t)\}_{K=1}^{\infty}$ to (4.5) with $s_{K+1}(t) = 0$.

Denote $s^K(t)$ as the solution to (4.5) with $s_{K+1}^K = 0$. Then we have $s_{K+1}^{K+1}(t) \geq s_{K+1}^K(t) = 0$. By the previous argument, we have that for fixed t and $i \leq K$, $s_i^{K+1}(t) \geq s_i^K(t)$. Then there exists the limit $\lim_{K\to\infty} s_i^K(t) = s_i(t)$ for each i and $s(t) = \{s_i(t)\}_{i=0}^{\infty} \in \bar{\mathcal{S}}$. Notice that $s_i(t)$ satisfies the fluid limit equation (4.5). It follows by uniqueness of the solution that the limit $\lim_{K\to\infty} s^K(t) = s(t)$ is the solution to fluid limit Equation (4.5). Finally, combining the two previous arguments, we conclude the comparison theorem for infinite dimensional case.

□

Theorem 13 (Global Stability of Fluid Limit)**.** *The fluid limit equation (4.5) has* *globally attractive stable point s^I. That is, starting from any initial condition $s(0) \in \mathcal{S}$,*

$$\lim_{t\to\infty} s(t) = s^I$$

Proof. It is sufficient to show that the conclusion $\lim_{t\to\infty} s(t) = s^I$ holds for any $s(0) \in \mathcal{S}$ for which either $s(0) \leq s^I$ or $s(0) \geq s^I$, since Lemma 5 implies that

$$s(t, \min[s(0), s^I]) \leq s(t, s(0)) \leq s(t, \max[s(0), s^I])$$

where we use $s(t, u)$ to denote the solution to Equation (4.5) with initial condition u.

Since the derivative of $s_k(t, s(0))$ is bounded for all k, the convergence of $s(t, s(0)) \to s^I$ will follow from

$$\int_0^\infty [s_k(u, s(0)) - s_k^I]du < \infty, \quad \text{where } s(0) \geq s^I \tag{4.12}$$

and from

$$\int_0^\infty [s_k^I - s_k(u, s(0))]du < \infty, \quad \text{where } s(0) \leq s^I.$$

The proof is similar for both cases so here we only discuss (4.12).

Define $v_k(s(t)) = \sum_{i=k}^\infty s_k(t)$, and $v(s) = \{v_i(s)\}_{i=0}^\infty$. Then we have for any $k \in \mathbb{N}$ and fix $t \geq 0$,

$$0 \leq v_k(s(t)) \leq v_1(s(t)) = \sum_{i=1}^\infty s_i(t) < \infty.$$

We also know that

$$\frac{d(v_1(s(t)) - v_1(s^I))}{dt} = \lambda(1-p)s_0 + \lambda p s_0^d - s_1(t) = \lambda - s_1(t) = s_1^I - s_1(t) \leq 0,$$

which implies that $v_1(s(t))$ does not increase with t. Thus, $v_1(s(t))$ is uniformly bounded for all $t \geq 0$. Notice that

$$\begin{aligned}\frac{dv_k(s(t))}{dt} &= \lambda(1-p)s_{k-1}(t) + \lambda p s_{k-1}(t)^d - s_k(t) \\ &= \lambda(1-p)(s_{k-1}(t) - s_{k-1}^I) + \lambda p((s_{k-1}(t))^d - (s_{k-1}^I)^d) - (s_k(t) - s_k^I),\end{aligned}$$

which implies that

$$\begin{aligned}& v_k(s(t)) - v_k(s(0)) \\ = & \int_0^t \left[\lambda(1-p)(s_{k-1}(u) - s_{k-1}^I) + \lambda p((s_{k-1}(u))^d - (s_{k-1}^I)^d) - (s_k(u) - s_k^I)\right] du.\end{aligned}$$

By the uniform boundedness of $v_k(s(t)) - v_k(s(0))$, we know that

$$\int_0^\infty \left[\lambda(1-p)(s_{k-1}(u) - s_{k-1}^I) + \lambda p((s_{k-1}(u))^d - (s_{k-1}^I)^d) - (s_k(u) - s_k^I)\right] du < \infty.$$

Using an induction argument, we can assume that the integral converges for all $i \leq k-1$, i.e.

$$\int_0^\infty (s_i(t) - s_i^I)dt < \infty, \quad i \leq k-1.$$

Then for $i = k$, again by the uniform boundedness of $v_k(s(t)) - v_k(s(0))$ we have that

$$\int_0^\infty (s_k(t) - s_k^I)dt < \infty,$$

which completes the proof of global stability of the fluid limit $\lim_{t\to\infty} s(t) = s^I$ for any initial condition $s(0)$. □

Now we will construct a coupling which compares the behavior of the system $S^N(t)$ when $d = 1$ vs. $d > 1$. When $d = 1$, the fluid limit equation becomes

$$\dot{s_i}(t) = \lambda(s_{i-1}(t) - s_i(t)) - (s_i(t) - s_{i+1}(t)),$$

which is a system of N i.i.d $M/M/1$ queues. And we know that if and only if $\lambda < 1$, when such system is positive recurrent, with a geometric stationary distribution being

$$s_k^I = \lambda^k, \quad k \in \mathbb{N}.$$

Let's consider coupling three systems with choices between 1 queue, d queues and with probability p of being flexible respectively, and we call them system 0, system 1 and system p. We use $\sigma = \{0, 1, p\}$ to denote quantities related to system σ by superscript σ. We use $c_m^{N,\sigma}(t)$ to denote the number of patients which have at least m patients queueing in front of then at time $t \geq 0$, which can be written as

$$c_m^{N,\sigma}(t) = N \sum_{k \geq m+1} S_k^{N,\sigma}(t), \quad m \in \mathbb{N}.$$

We will first focus on comparing system 0 and system p. We use a single Poisson process of rate $N\lambda$ for arrivals for both systems. At each jump time, we generate a random variable with $Bernoulli(p)$ distribution to decide whether the patient is flexible or not. If he/she is flexible, we choose uniformly $j_1^p < \cdots < j_d^p$ among $1, \cdots, N$ and then j^0 among $j_1^p < \cdots < j_d^p$, and set $j^p = j_d^p$. If the patient is not flexible, we simply choose uniformly j among $\{1, \cdots, N\}$ and set $j^p = j^0 = j$. In system σ, we order the queues by decreasing length (ties are resolved with uniform probability), and let the task join the queue ranking j^σ in this order. Note that $j^0 \leq j^p$.

We use a single Poisson process of rate N for potential departures for both systems. At each jump time, we choose j uniformly in $\{1, \cdots, N\}$. In system σ, we again order the queues by decreasing length, and remove a task from the j^{th} queue in this order if that queue is not empty.

Our goal is to show that performance is ranked as follows (system 1 $\leq$ system $p \leq$ system 0) with respect to the number of patients in the system. Our proof of this coupling is a modification of the proof given in Theorem 4 of Turner [82].

Proposition 6 (Coupling Result). *For $N \in \mathbb{N}$, if $c_m^{N,1}(0) \leq c_m^{N,p}(0) \leq c_m^{N,0}(0)$ for all $m \in \mathbb{N}$, then*

$$c_m^{N,1}(t) \leq c_m^{N,p}(t) \leq c_m^{N,0}(t), \quad m \in \mathbb{N}, t \geq 0.$$

Proof. Let τ be a jump time of the Poisson processes used for arrivals and departures. We first compare system p with system 0. Our goal is to show that assuming

$$c_m^{N,p}(\tau-) \leq c_m^{N,0}(\tau-), \quad m \in \mathbb{N}, \tag{4.13}$$

then we have

$$c_m^{N,p}(\tau-) \leq c_m^{N,0}(\tau), \quad m \in \mathbb{N}.$$

Since we know that

$$c_m^{N,\sigma}(t) = c_{m+1}^{N,\sigma}(t) + NS_{m+1}^{N,\sigma}(t), \quad m \geq 0, t \geq 0.$$

Applying (4.20) to $m = n-1$ and $m = n$ implies

$$c_n^{N,1}(\tau-) = c_n^{N,0}(\tau-) \Rightarrow S_n^{N,1}(\tau-) \leq S_n^{N,0}(\tau-) \text{ and } S_{n+1}^{N,1}(\tau-) = S_{n+1}^{N,0}(\tau-).$$

When τ represent a departure time, let x^σ denote the respective lengths of the queue chosen for potential departure. A patient will depart from the system σ if and only if $x^\sigma > 0$, and there will be one less patient with exactly $x^\sigma - 1$ patients in front of them, therefore

$$c_m^{N,\sigma}(\tau) = c_m^{N,\sigma}(\tau-) - 1, \quad m < x^\sigma, \tag{4.14}$$

and

$$c_m^{N,\sigma}(\tau) = c_m^{N,\sigma}(\tau-), \quad m \geq x^\sigma. \tag{4.15}$$

Assume that there exists $n \geq 0$ such that $c_n^{N,p}(\tau) > c_n^{N,0}(\tau)$, then (4.20), (4.14) and (4.15) imply that it is true if and only if

$$c_n^{N,p}(\tau) = c_n^{N,1}(\tau), \quad x^p \leq n < x^0. \tag{4.16}$$

Now let $j \in \{1, \cdots, N\}$ denotes the rank in decreasing order chosen for departures, then

$$NS_{x^\sigma+1}^{N,\sigma}(\tau-) < j \leq NS_{x^\sigma}^{N,\sigma}(\tau-)$$

which yields in particular that

$$S_{x^1+1}^{N,p}(\tau-) < S_{x^1}^{N,p}(\tau-) \leq S_{x^0}^{N,0}(\tau-).$$

Then combining (4.14), (4.15) and (4.16) yields

$$S_{n+1}^{N,0}(\tau-) \leq S_{n+1}^{N,p}(\tau-) \leq S_{x^1+1}^{N,p}(\tau-) < S_{x^0}^{N,0}(\tau-) \leq S_{n+1}^{N,0}(\tau-)$$

which is a contradiction. Thus $c_m^{N,p}(\tau) \leq c_m^{N,0}(\tau), \quad m \in \mathbb{N}$ holds.

When τ represent an arrival time, let x^σ denote the respective lengths of the queues chosen for either patient. There is a new patient in either system with x^σ patients in front of him, therefore

$$c_m^{N,\sigma}(\tau) = c_m^{N,\sigma}(\tau-) + 1, \quad m \leq x^\sigma \tag{4.17}$$

and

$$c_m^{N,\sigma}(\tau) = c_m^{N,\sigma}(\tau-), \quad m > x^\sigma \tag{4.18}$$

Assume that there exists $n \geq 0$ such that $c_n^{N,p}(\tau) > c_n^{N,0}(\tau)$, then (4.20), (4.17) and (4.18) imply that it is true if and only if

$$c_n^{N,0}(\tau) = c_n^{N,p}(\tau), \quad x^0 \leq n < x^p \tag{4.19}$$

Now let $j^\sigma \in \{1, \cdots, N\}$ denotes the rank in decreasing order of the queue joined by the patient in system σ, then

$$NS^{N,\sigma}_{x^\sigma+1}(\tau-) < j^\sigma \leq NS^{N,\sigma}_{x^\sigma}(\tau-)$$

which yields in particular that

$$S^{N,0}_{x^0+1}(\tau-) < j^0 \leq j^p \leq S^{N,p}_{x^p}(\tau-).$$

Then combining (4.17), (4.18) and (4.19) yields

$$S^{N,p}_n(\tau-) \leq S^{N,0}_n(\tau-) \leq S^{N,0}_{x^0+1}(\tau-) < S^{N,p}_{x^p}(\tau-) \leq S^{N,p}_n(\tau-)$$

which is a contradiction. Thus $c^{N,p}_m(\tau) \leq c^{N,0}_m(\tau), \quad m \in \mathbb{N}$ holds.

Similar techniques apply to the case of comparing system 0 and system p, and we have that if

$$c^{N,1}_m(\tau-) \leq c^{N,p}_m(\tau-), \quad m \in \mathbb{N}, \tag{4.20}$$

then

$$c^{N,1}_m(\tau-) \leq c^{N,p}_m(\tau), \quad m \in \mathbb{N}.$$

□

Theorem 14 (Convergence of Stationary Distributions)**.**

1. *The Markov process $S^N(t)$ is positive recurrent for all N, and therefore has a unique stationary distribution $\pi^N \in \mathcal{P}(\bar{S})$ for each N.*
2. *The sequence of stationary distribution π^N of process $S^N(t)$ converges weakly to the Dirac mass at s^I as $N \to \infty$.*

Proof. By Theorem 6, the system 1 is empty whenever system 0 is. Therefore system 1 is also positive recurrent when $\lambda < 1$ and have a stationary distribution π^N. Irreducibility implies the uniqueness of the stationary distribution.

Since $\bar{S}$ is compact, so is the set $\mathcal{P}(\bar{S})$ of the probability measures on $\bar{S}$. Therefore the sequence of probability measures $\{\pi^N\}_{N=1}^{\infty}$ is tight and has limit points. We aim to show that any limit point of $\{\pi^N\}_{N=1}^{\infty}$ is the Dirac mass at s^I.

Assume that $S^N(0)$ has the same distribution as the stationary distribution π^N, for each N. By Theorem 11, let $\pi^\infty(0)$ be the limiting distribution of a subsequence of $(S^N(0))_{N\geq 1}$, and let $\pi^\infty(t)$ be the limiting distribution for the same subsequence of $(S^N)_{N\geq 1}$. For $t \geq 0$ and $N \geq 1$, since the process started with its stationary distribution, we have that $S^N(t)$ also follows distribution π^N. Applying Theorem 11, we have that the fluid limit $s(t) = \lim_{N\to\infty} S^N(t)$ has the same distribution as $\pi^\infty(0)$.

Now let $\epsilon > 0$ and V be an open neighborhood of s^I. For $j \in \mathbb{N}$, let P_j be the set of all a in $\mathcal{P}(S)$ such that the solution for the (4.5) starting at a is in V for all times $t \geq j$. Since P_j is measurable, $P_j \subset P_{j+1}$, and by the fact that s^I is a globally attractive point (Theorem 13), we have $\mathcal{P}(S) = \cup_j P_j$, therefore there exists k such that $P(\pi^\infty(0) \in P_k) > 1 - \epsilon$. Then

$$P(\pi^\infty(0) \in V) = P(\pi^\infty(k) \in V) \geq P(\pi^\infty(0) \in P_k) > 1 - \epsilon.$$

Since ϵ and V arbitrary, we have $P(\pi^\infty(0) = s^I) = 1$. Hence $(S^N(0))_{N\geq 1}$ converges in distribution to the Dirac mass at s^I, and the limiting distribution of $(S^N)_{N\geq 1}$ is the constant s^I.

□

4.3 Diffusion Model

In this section, we analyze a diffusion scaled version of the queueing process. Since the fluid limit does not capture stochastic fluctuations, the diffusion model can help us gain important insights on the fluctuations of the system, which can be used to build confidence intervals for various performance measures. To do this, we first prove a functional central limit theorem (CLT) in the transient case for the scaled diffusion process $D^N(t) = \sqrt{N}(S^N(t) - s(t))$ to its limit $D(t)$. We identify $D(t)$ as an infinite dimensional Ornstein Uhlenbeck (OU) process. By computing the variance of $D(t)$, we can construct rigorous confidence intervals for characterizing the deviations from the fluid limit in the transient setting. Second, we prove the functional CLT in the equilibrium setting, thereby establishing an interchange of limits result for the diffusion scaled empirical process. We prove the interchange by showing convergence in the appropriate Hilbert spaces and deriving novel infinite horizon bounds for the diffusion scaled process.

4.3.1 Transient Analysis

In this section, we derive the diffusion limit of our stochastic queueing model in the transient setting. We define our scaled diffusion process as

$$D^N(t) = \sqrt{N}(S^N(t) - s(t)).$$

Theorem 15 (Functional Central Limit Theorem)**.** *Consider ℓ_2 with its weak topology and $\mathbb{D}(\mathbb{R}_+, \ell_2)$ with corresponding Skorokhod topology. Let $s(0)$ be in $\mathcal{S} \bigcap \ell_1$, $S^N(0)$ in $\mathcal{S}^N$. If $(D^N(0))_{N \geq d}$ converges in distribution to $D(0)$ and is tight, then $(D^N(t))_{N \geq d}$*

is tight and converges in distribution to the unique OU process

$$D(t) = D(0) + \int_0^t K(s(u))D(u)du + M(t)$$

where the infinite dimensional matrix $K(s)$ is given by

$$\begin{aligned} K_{i,i}(s) &= -\lambda(1-p) - \lambda p d s_i^{d-1} - 1, \\ K_{i,i+1}(s) &= 1, \\ K_{i+1,i}(s) &= \lambda(1-p) + \lambda p d s_i^{d-1} \end{aligned}$$

for $i \in \mathbb{Z}_+$ and the martingale $M(t)$ is defined by the Doob-Meyer brackets

$$< M_k(t) > \quad = \quad \int_0^t \left[F_+(s(u))(k) + F_-(s(u))(k)\right] du.$$

Consider a linearization of Equation (4.5) around a particular solution g such that

$$d(t) = g(t) - s(t),$$

where g is a generic solution to Equation (4.5). Then we have

$$\dot{d}(t) = K(s(t))d(t), \tag{4.21}$$

where K is a matrix in $\mathbb{Z}_+ \times \mathbb{Z}_+$ with entries

$$\begin{aligned} K_{i,i}(s) &= -\lambda(1-p) - \lambda p d s_i^{d-1} - 1, \\ K_{i,i+1}(s) &= 1, \\ K_{i+1,i}(s) &= \lambda(1-p) + \lambda p d s_i^{d-1} \end{aligned}$$

for $i \in \mathbb{Z}_+$.

Let $(M_k(t))_{k\in\mathbb{N}}$ be a family of independent, real, continuous centered Gaussian martingales, determined in law by their deterministic Doob-Meyer brackets given by

$$\begin{aligned} & < M_k(t) > \\ = & \int_0^t \left[\lambda(1-p)(s_{i-1}(u) - s_i(u)) + \lambda p \left(s_{i-1}(u)^d - s_i(u)^d\right) + (s_i(u) - s_{i+1}(u))\right] du \\ = & \int_0^t \left[F_+(s(u))(k) + F_-(s(u))(k)\right] du. \end{aligned}$$

for $t \geq 0$.

To give readers a high-level understanding of the proof idea, we summarize the following list of steps for showing the functional CLT in the transient case,

1. Prove the Lipschitz property for the mappings F, F_+, F_- in ℓ_2. (Theorem 4)
2. Prove the Gaussian martingale $M(t)$ is square-integrable in ℓ_2. (Theorem 16)
3. Prove the existence and uniqueness of the diffusion limit $D(t)$ by using steps 1 and 2 to show that Equation (4.23) is well-defined and solves the SDE.
4. Show the difference between the drift function $F^N(s)$ and the limiting drift function $F(s)$ is $\frac{1}{N}O(s)$. (Lemma 8)
5. Show the finite horizon bound
$$\limsup_{N\to\infty} \mathbb{E}\left(\|D^N(0)\|_{\ell_2}^2\right) < \infty \Rightarrow \limsup_{N\to\infty} \mathbb{E}\left(\sup_{t\leq T} \|D^N(t)\|_{\ell_2}^2\right) < \infty$$
using Doob's inequality, Gronwall's lemma and steps 1,2, and 4 .
6. Use step 5 to show the tightness of the diffusion process. (Lemma 10).
7. Use steps 1-6 to show the functional CLT, i.e. when initial condition converges, the diffusion process D^N converges to the unique OU process solving Equation (4.23). (Theorem 15)

Theorem 16. *Assume $s(0)$ to be in $\mathcal{S}$. Then, the Gaussian martingale $M(t)$ is square-integrable in ℓ_2.*

Proof. Because of the Lipschitz property of mappings F_+, F_-, we have

$$\|M(t)\|_{\ell_2} = \int_0^t \|F_+(s(u)) + F_-(s(u))\|_{\ell_2} du \leq \int_0^t (2\sqrt{2}\lambda d + \sqrt{2})\|s(u)\|_{\ell_2} du$$

By Gronwall's lemma, we know that $\|s(u)\|_{\ell_2}$ is uniformly bounded on $0 \leq u \leq t$. Thus, $M(t)$ is square-integrable in ℓ_2. $\square$

Let $D(t)$ be the diffusion limit for the fluctuations $D^N(t)$, which is a Gaussian perturbation of Equation (4.21), then $D(t)$ satisfies the following SDE for any given $t \geq 0$,

$$D(t) = D(0) + \int_0^t K(s(u))D(u)du + M(t). \tag{4.22}$$

Theorem 17 (Existence and Uniqueness of Diffusion Limit)**.**

1. *For s in $\mathcal{S}$, the operator $K(s)$ is bounded in ℓ_2 with operator norm uniformly bounded in s.*

2. *Let $s(0)$ be in $\mathcal{S} \bigcap \ell_1$. Then there exists a unique strong solution to Equation (4.22) in ℓ_2*

$$D(t) = \exp\left\{\int_0^t K(s(u))du\right\} D(0) + \int_0^t \exp\left\{\int_u^t K(s(r))dr\right\} dM(u), \tag{4.23}$$

and

$$\mathbb{E}\left(\|D(0)\|_{\ell_2}^2\right) < \infty \Rightarrow \mathbb{E}\left(\sup_{t \leq T} \|D(t)\|_{\ell_2}^2\right) < \infty.$$

Proof. Consider $s \in \mathcal{S}$, we have

$$\begin{aligned} & \|K(s)x\|_{\ell_2}^2 \\ = & \sum_{k\geq 1} \left[\lambda(1-p)(x_{k-1}-x_k) + \lambda pd\left(s_{k-1}^{d-1}x_{k-1} - s_k^{d-1}x_k\right) + x_k - x_{k+1}\right]^2 \\ \leq & \sum_{k\geq 1} \left((\lambda(1-p)+\lambda pd)^2 + (\lambda(1-p)+\lambda pd+1)^2+1)^2+1^2\right)\left(x_{k-1}^2+x_k^2+x_{k+1}^2\right) \\ \leq & \; 6(\lambda(1-p)+\lambda pd+1)^2\|x\|_{\ell_2}^2. \end{aligned}$$

Then (1) follows. For (2), since the martingale $M(t)$ is square-integrable in ℓ_2 by Theorem 16, if $\mathbb{E}\left(\|D(0)\|_{\ell_2}^2\right) < \infty$, then the formula (4.23) for $D(t)$ is well-defined, solves the SDE, and using Gronwall's lemma yields $\mathbb{E}\left(\sup_{t\leq T}\|D(t)\|_{\ell_2}^2\right) < \infty$. □

For the following Lemma 6 and Theorem 15, the proofs are detailed in subsections 4.3.1 and 4.3.1.

Lemma 6 (Finite Horizon Bound). *Let $s(0)$ be in $\mathcal{S}\bigcap\ell_1$ and $S^N(0)$ be in $\mathcal{S}^N$. Then for any $T \geq 0$,*

$$\limsup_{N\to\infty}\mathbb{E}\left(\|D^N(0)\|_{\ell_2}^2\right) < \infty \Rightarrow \limsup_{N\to\infty}\mathbb{E}\left(\sup_{t\leq T}\|D^N(t)\|_{\ell_2}^2\right) < \infty.$$

Theorem 18. *Define the two matrices $\mathcal{A}(t) = K(s(t))$, $\mathcal{B}(t) = \left(\frac{d}{dt}\langle M_i(t), M_j(t)\rangle\right)_{ij}$, then the expectation $E(D(t))$ is*

$$\mathbb{E}[D(t)] = e^{\int_0^t \mathcal{A}(s)ds}\mathbb{E}[D(0)],$$

and the covariance matrix $\Sigma(t) = \mathrm{Cov}[D(t), D(t)]$ is

$$\Sigma(t) = e^{\int_0^t \mathcal{A}(s)ds}\Sigma(0)e^{\int_0^t \mathcal{A}^\top(s)ds} + \int_0^t e^{\int_s^t \mathcal{A}(u)du}\mathcal{B}(s)e^{\int_s^t \mathcal{A}^\top(u)du}ds.$$

Moreover, differentiation with respect to t yields

$$\frac{d\mathbb{E}[D(t)]}{dt} = \mathcal{A}(t)\mathbb{E}[D(t)],$$

$$\frac{d\Sigma(t)}{dt} = \Sigma(t)\mathcal{A}(t)^\top + \mathcal{A}(t)\Sigma(t) + \mathcal{B}(t). \tag{4.24}$$

Componentwise, we have

$$\begin{aligned}\frac{d\Sigma_{i,i}(t)}{dt} &= 2\left[\lambda(1-p) + \lambda p d s_{i-1}^{d-1}\right]\Sigma_{i,i-1} - 2\left[\lambda(1-p) + \lambda p d s_i^{d-1} + 1\right]\Sigma_{i,i} + 2\Sigma_{i,i+1} \\ &\quad +\lambda(1-p)(s_{i-1} - s_i) + \lambda p(s_{i-1}^d - s_i^d) + s_i - s_{i+1},\end{aligned}$$

$$\begin{aligned}\frac{d\Sigma_{i,j}(t)}{dt} &= \left[2\lambda(1-p) + \lambda p d(s_{i-1}^{d-1} + s_{j-1}^{d-1})\right]\Sigma_{j,i-1} \\ &\quad - \left[2\lambda(1-p) + \lambda p d(s_i^{d-1} + s_j^{d-1}) + 2\right]\Sigma_{j,i} + 2\Sigma_{j,i+1}.\end{aligned}$$

Proof. Take expectation on both sides of Equation (4.23), since

$$\mathbb{E}\left[\int_0^t e^{\int_s^t K(s(u))du} dM(s)\right] = 0,$$

we have

$$\mathbb{E}[D(t)] = e^{\int_0^t \mathcal{A}(s)ds}\mathbb{E}[D(0)].$$

Therefore

$$D(t) - \mathbb{E}[D(t)] = e^{\int_0^t \mathcal{A}(s)ds}\left(D(0) - \mathbb{E}[D(0)]\right) + \int_0^t e^{\int_s^t \mathcal{A}(u)du} dM(s),$$

and

$$\begin{aligned}\Sigma(t) &= \mathbb{E}\left[(D(t) - \mathbb{E}[D(t)])(D(t) - \mathbb{E}[D(t)])^\top\right] \\ &= e^{\int_0^t \mathcal{A}(s)ds}\mathbb{E}\left[(D(0) - \mathbb{E}[D(0)])(D(0) - \mathbb{E}[D(0)])^\top\right]\left(e^{\int_0^t \mathcal{A}(s)ds}\right)^\top \\ &+ \left(\int_0^t e^{\int_s^t \mathcal{A}(u)du} dM(s)\right)\left(\int_0^t e^{\int_s^t \mathcal{A}(u)du} dM(s)\right)^\top \\ &= e^{\int_0^t \mathcal{A}(s)ds}\Sigma(0)e^{\int_0^t \mathcal{A}^\top(s)ds} + \int_0^t e^{\int_s^t \mathcal{A}(u)du}\mathcal{B}(s)e^{\int_s^t \mathcal{A}^\top(u)du}ds.\end{aligned}$$

□

The Derivation of the Ornstein-Uhlenbeck Process

We introduce a few lemmas which help show the final functional CLT result in the transient case.

Lemma 7. *Let $S^N(0)$ be in $\mathcal{S}^N$, s solves Equation (4.5) with $s(0) \in \mathcal{S}$. Then*

$$D^N(t) = D^N(0) + \int_0^t \sqrt{N}(F^N(S^N(u)) - F(s(u)))du + M^N(t) \tag{4.25}$$

defines an independent family of square-integrable martingales M^N independent of $S^N(0)$ with Doob-Meyer brackets given by

$$< M_k^N(t) >= \int_0^t \left(F_+^N(S^N(u)(k)) + F_-(S^N(u))(k)\right) du.$$

Proof. This follows from a classical application of Dynkin's formula. □

Lemma 8. *Define function $A^N(a)$ for $a \in \mathbb{R}$ and $N \geq d \geq 1$ as*

$$A^N(a) \triangleq \frac{(Na)_d}{(N)_d} - a^d.$$

Then, $A^N(a) = \frac{1}{N}O(a)$ uniformly on $0 \leq a \leq 1$ and $A^N(k/N) \leq 0$ for $k = 0, 1, \cdots, N$.

Proof. Since

$$\begin{aligned}
\frac{(Na)_d}{(N)_d} &= \prod_{i=0}^{d-1} \frac{Na - i}{N - i} \\
&= \prod_{i=0}^{d-1} \left(a + (a-1)\frac{i}{N-i}\right) \\
&= \sum_{j=1}^{d-1} a^{d-j}(a-1)^j \prod_{1 \leq i_1 < \cdots < i_j \leq d-1} \frac{i_1 \cdots i_j}{(N-i_1)\cdots(N-i_j)}
\end{aligned}$$

It is obvious that $A^N(a)$ is $\frac{1}{N}O(a)$. For $a = \frac{k}{N}$ where $k = 0, 1, \cdots, N$,

$$\begin{aligned}\prod_{i=0}^{d-1} \frac{Na-i}{N-i} &= \prod_{i=0}^{d-1} \frac{k-i}{N-i} \\ &\leq \prod_{i=0}^{d-1} \frac{k}{N} = a^d\end{aligned}$$

The inequality comes from the fact that each term $\frac{k-i}{N-i}$ is either bounded by a or the product contains a term exactly equal to 0. Thus $A^N(k/N) \leq 0$. □

Lemma 9. *For $d \geq 1$ and $a, h \in \mathbb{R}$, define*

$$B(a,h) \triangleq (a+h)^d - a^d - da^{d-1}h = \sum_{i=2}^{d} \binom{d}{i} a^{d-i} h^i.$$

Then $B(a,h) = 0$ for $d = 1$ and $B(a,h) = h^2$ for $d = 2$. For $d \geq 2$ we have $0 \leq B(a,h) \leq h^d + (2^d - d - 2)ah^2$ for $a, a+h \in [0,1]$.

Proof. For $a, a+h \in [0,1]$,

$$B(a,h) \leq h^d + \sum_{i=2}^{d-1} ah^2 = h^d + (2^d - d - 2)ah^2.$$

□

Proof of the functional CLT Consider the mapping $G^N : \mathcal{S} \to c_0^0$ given by

$$G^N(s)(k) = \lambda p \left(A^N(s_{k-1}) - A^N(s_k) \right), \quad k \geq 1$$

and $H : \mathcal{S} \times c_0^0 \to c_0^0$ given by

$$H(s,x)(k) = \lambda p (B(s_{k-1}, x_{k-1}) - B(s_k, x_k)), \quad k \geq 1$$

so that for $s + x \in \mathcal{S}$, we have

$$F^N = F + G^N, \quad F(s+x) - F(s) = K(s)x + H(s,x). \tag{4.26}$$

Proof of Lemma 6 (Finite-horizon bound). By Equations (4.25) and (4.26), we have

$$D^N(t) = D^N(0)+M^N(t)+\sqrt{N}\int_0^t G^N(S^N(u))du+\int_0^t \sqrt{N}(F(S^N(u))-F(s(u)))du. \tag{4.27}$$

Since Lemma 8 indicates that

$$\begin{aligned} G^N(S^N(u))(k) &= \lambda p \left(A^N(S^N(u)(k-1)) - A^N(S^N(u)(k)\right) \\ &= \frac{1}{N} O\left(S^N(u)(k-1) + S^N(u)(k)\right), \end{aligned}$$

we can conclude that

$$\|G^N(S^N(u))\|_{\ell_2} = \frac{1}{N} O\left(\|S^N(u)\|_{\ell_2}\right). \tag{4.28}$$

By definition of the diffusion process we have

$$\|S^N(u)\|_{\ell_2} \leq \|s(u)\|_{\ell_2} + \frac{1}{\sqrt{N}}\|D(u)^N\|_{\ell_2}. \tag{4.29}$$

Since mappings F_+, F_-, F are Lipschitz with respect to ℓ_2 norm, Gronwall's lemma yields that

$$\|s(u)\|_{\ell_2} \leq L_T\|s(0)\|_{\ell_2} \tag{4.30}$$

for some constant $L_T < \infty$. Then

$$\begin{aligned} \|D^N(t)\|_{\ell_2} &\leq \|D^N(0)\|_{\ell_2} + \|M^N(t)\|_{\ell_2} + \sqrt{N}\int_0^t \|G^N(S^N(u))\|_{\ell_2} du \\ &\quad + \int_0^t \sqrt{N}(\|F(S^N(u)) - F(s(u))\|_{\ell_2}) du \\ &\leq \|D^N(0)\|_{\ell_2} + \|M^N(t)\|_{\ell_2} + \sqrt{N}\int_0^t \frac{1}{N} O\left(\|S^N(u)\|_{\ell_2}\right) du \\ &\quad + \int_0^t \sqrt{N} L(\|S^N(u) - s(u)\|_{\ell_2}) du \\ &\leq \|D^N(0)\|_{\ell_2} + \|M^N(t)\|_{\ell_2} + \int_0^t \frac{1}{\sqrt{N}} O\left(\|s(u)\|_{\ell_2} + \frac{1}{\sqrt{N}}\|D(u)^N\|_{\ell_2}\right) du \\ &\quad + \int_0^t L(\|D(u)^N\|_{\ell_2}) du \\ &\leq \|D^N(0)\|_{\ell_2} + \|M^N(t)\|_{\ell_2} + \frac{1}{\sqrt{N}} O\left(L_T\|s(0)\|_{\ell_2}\right) \\ &\quad + \int_0^t \left(L + O\left(\frac{1}{N}\right)\right) \|D(u)^N\|_{\ell_2} du. \end{aligned}$$

By Gronwall's lemma we have

$$\begin{aligned} &\sup_{0\leq t\leq T}\|D^N(t)\|_{\ell_2} \\ \leq\ & \exp\left\{\left(L+O\left(\frac{1}{N}\right)\right)T\right\}\left(\|D^N(0)\|_{\ell_2}+\sup_{0\leq t\leq T}\|M^N(t)\|_{\ell_2}+\frac{L_T}{\sqrt{N}}O\left(\|s(0)\|_{\ell_2}\right)\right). \end{aligned} \tag{4.31}$$

Using Doob's ℓ_2 inequality we know that,

$$\mathbb{E}\left(\sup_{0\leq t\leq T}\|M^N(t)\|_{\ell_2}\right)\leq 2\mathbb{E}\left(\|M^N(T)\|_{\ell_2}\right).$$

By Lemma 7 and Lipschitz property of F_+, F_-,

$$\begin{aligned} \|M_T^N\|_{\ell_2} &= \int_0^T \|F_+^N(S^N(u))+F_-(S^N(u))\|_{\ell_2}du \\ \text{(Equation (4.26))} &= \int_0^T \|F_+(S^N(u))+G^N(S^N(u))+F_-(S^N(u))\|_{\ell_2}du \\ \text{(Equation (4.28))} &\leq \int_0^T\left(2L\|S^N(u)\|_{\ell_2}+\frac{1}{N}O(\|S^N(u)\|_{\ell_2})\right)du \\ \text{(Equation (4.29))} &\leq \int_0^T O\left(\|s(u)\|_{\ell_2}+\frac{1}{\sqrt{N}}\|D(u)^N\|_{\ell_2}\right)du \\ \text{(Equation (4.30))} &= K_T O(\|s(0)\|_{\ell_2}). \end{aligned}$$

Finally combining all the above equations, we conclude that when

$$\limsup_{N\to\infty}\mathbb{E}\left(\|D^N(0)\|_{\ell_2}^2\right)<\infty,$$

we have

$$\begin{aligned} &\limsup_{N\to\infty}\mathbb{E}\left(\sup_{0\leq t\leq T}\|D^N(t)\|_{\ell_2}\right) \\ \leq\ & \exp\{O(T)\} \\ &\cdot\left(\limsup_{N\to\infty}\mathbb{E}\left(\|D^N(0)\|_{\ell_2}\right)+\limsup_{N\to\infty}\mathbb{E}\left(\sup_{0\leq t\leq T}\|M^N(t)\|_{\ell_2}\right)+\frac{L_T}{\sqrt{N}}O\left(\|s(0)\|_{\ell_2}\right)\right) \\ \leq\ & \exp\{O(T)\} \\ &\cdot\left(\limsup_{N\to\infty}\mathbb{E}\left(\|D^N(0)\|_{\ell_2}\right)+\limsup_{N\to\infty}2\mathbb{E}\left(\|M^N(T)\|_{\ell_2}\right)+\frac{L_T}{\sqrt{N}}O\left(\|s(0)\|_{\ell_2}\right)\right) \\ \leq\ & \exp\{O(T)\}\left(\limsup_{N\to\infty}\mathbb{E}\left(\|D^N(0)\|_{\ell_2}\right)+\left(2K_T+\frac{L_T}{\sqrt{N}}\right)O\left(\|s(0)\|_{\ell_2}\right)\right) \\ <\ & \infty. \end{aligned}$$

□

Lemma 10 (Tightness of the Process)**.** *Consider ℓ_2 with its weak topology and $\mathbb{D}(\mathbb{R}_+;\ell_2)$ with the corresponding Skorokhod topology. Assume $s(0) \in \mathcal{S} \cap \ell_1$ and $S^N(0) \in \mathcal{S}^N$, and D^N as defined in the beginning of the section. If $(D^N(0))_{N\geq d}$ is tight, then $(D^N)_{N\geq d}$ is tight and its limit points are continuous.*

Proof. Since $\mathbb{D}(\mathbb{R}_+;\ell_2)$ is a reflexive Banach space, relatively compact sets are the bounded sets for the norm ℓ_2. Then here a process D^N is tight if and only if for any $\epsilon > 0$ there exists $r_\epsilon < \infty$ such that $\mathbb{P}(D^N \in B(r_\epsilon)) > 1-\epsilon$ for $N \geq 1$. We refer to Ethier and Kurtz [31] the tightness criteria for showing that $(D^N)_{N\geq d}$ is tight. That is, $(D^N)_{N\geq d}$ is tight if

1. For each $T \geq 0$ and $\epsilon > 0$ there is a bounded subset $K_{T,\epsilon} \in \ell_2$ such that $\mathbb{P}(D^N \in D([0,T];K_{T,\epsilon})) > 1-\epsilon$ for $N \geq d$.
2. For each $k \geq 1$, the k-dimensional process $(D_1^N, D_2^N, \cdots, D_k^N)_{N\geq d}$ are tight.

For Condition 1, it is easy to see that using finite-horizon bound in Lemma 6 and Markov inequality, we can derive the tightness of process D^N on $D([0,T];K_{T,\epsilon})$.

For Condition 2, we refer to Graham [39] for the fact that bounds in Lemma 6 and that D_k^N has jump size of $\frac{1}{\sqrt{N}}$ classically imply the tightness of the finite-dimensional process. □

Proof of Theorem 15 (Functional CLT). Using Lemma 10, we know that any subsequence of D^N has a further subsequence that converges to some limit D^∞ with continuous sample path. $D^\infty(0)$ should have the same distribution as $D(0)$. We can rewrite Equation (4.27) as

$$\begin{aligned} D^N(t) &= D^N(0) + M^N(t) + \int_0^t K(s(u))D^N(u)du \\ &+\sqrt{N}\int_0^t \left(G^N(S^N(u)) + H\left(s(u), D^N(u)/\sqrt{N}\right)\right) du. \end{aligned}$$

Using Equations (4.28), (4.29), we have that

$$\sqrt{N}\|G^N(S^N(u))\|_{\ell_2} = \frac{1}{\sqrt{N}}O(\|S^N(u)\|_{\ell_2}) \to 0$$

as $N \to \infty$. Using Lemma 9, we have

$$\begin{aligned} &\sqrt{N}\|H(s(u), D^N(u)/\sqrt{N})\|_{\ell_2} \\ \leq\ &\sqrt{N}\lambda p\left[\frac{1}{N^{d/2}}\|(D^N(u))^d\|_{\ell_2} + \frac{1}{N}(2^d - d - 2)\|s(u)\|_{\ell_2} \cdot \|D^N(u)\|_{\ell_2}^2\right] \\ \to\ &0 \end{aligned}$$

as $N \to \infty$. We also have the martingale brackets

$$\begin{aligned} < M_k^N(t) > &= \int_0^t \left(F_+^N(S^N(u))(k) + F_-(S^N(u))(k)\right) du \\ &\to \int_0^t (F_+(s(u))(k) + F_-(s(u))(k))\, du \\ &= < M_k(t) > \end{aligned}$$

as $N \to \infty$.

By Theorem 4.1 in Ethier and Kurtz [31], together with Lipschitz property of F in Lemma 4, finite horizon bounds in Lemma 6 and tightness results in Lemma 10, we deduce by a martingale characterization that D^∞ has the distribution of the OU process which is the unique solution for (4.22) in ℓ_2 starting at $D^\infty(0)$. Thus, this distribution D^∞ is the unique accumulation point for the relatively compact sequence of distributions of $(D^N)_{N\geq 1}$, therefore itself must then converge to it, proving Theorem 15. □

4.3.2 Steady State Analysis

In this section, we analyze the steady state of the diffusion model. This allows us to gain insights about the long-time behavior of the nonlinear system dynamics appearing at the large N limit. Assume we have $\lambda < 1$, and that $s(0) = s^I$. Define the infinite-dimensional matrix $\mathcal{K} = K(s^I)$. Then we have

$$\begin{aligned} \mathcal{K}_{i,i}(s) &= -\lambda(1-p) - \lambda pd(s_i^I)^{d-1} - 1, \\ \mathcal{K}_{i,i+1}(s) &= 1, \\ \mathcal{K}_{i+1,i}(s) &= \lambda(1-p) + \lambda pd(s_i^I)^{d-1} \end{aligned}$$

for $i \in \mathbb{Z}_+$.

Note that $\mathcal{K} = \mathcal{A}^*$ where $\mathcal{A}$ is the generator of a sub-Markovian birth-death process. We use $\pi = (\pi_k)_{k\geq 1}$ to denote the tail cdf of the the stationary distribution to $\mathcal{A}$. Then, π solves the following balance equations

$$\begin{aligned} \pi_1 &= 1, \\ \pi_{k+1} &= \left[\lambda(1-p) + \lambda pd(s_k^I)^{d-1}\right]\pi_k, \quad k \geq 1. \end{aligned}$$

Consider the independent and centered Brownian motions $B(t) = (B_k(t))_{k\geq 0}$ such that $B(0) = 0$, and for $k \geq 1$

$$v_k \triangleq \text{Var}(B_k(1)) = \mathbb{E}(B_k(1)) = 2(s_k^I - s_{k+1}^I)$$

and B has an infinitesimal covariance matrix diag(v). The OU process $D(t) = (D_k(t))_{k\in\mathbb{N}}$ solves the affine SDE given for $t \geq 0$ by

$$D(t) = D(0) + \int_0^t \mathcal{K}D(s)ds + B(t) \tag{4.32}$$

which is a Brownian perturbation of the following differential equation

$$\dot{d}(t) = \mathcal{K}d(t). \tag{4.33}$$

Our ultimate goal is to show the interchanging of limits for the diffusion model. However, a main difficulty is that the scalar product for which the operator $\mathcal{K}$ is self-adjoint is too strong for the limit dynamical system and the invariant measures for finite N. Thus, we need to consider appropriate Hilbert spaces in which the operator $\mathcal{K}$ is not self-adjoint and prove the exponential stability of the fluid limit in the newly introduced space. As a result, we introduce the following weighted Hilbert space

$$L_2(w) \triangleq \left\{ x \in \mathbb{R}^{\mathbb{N}} : x(0) = 0, \|x\|^2_{L_2(w)} = \sum_{k\geq 1} x(k)^2 w(k)^{-1} < \infty \right\}.$$

We also consider the following ℓ_1 space with same weights

$$L_1(w) \triangleq \left\{ x \in \mathbb{R}^{\mathbb{N}} : x(0) = 0, \|x\|^2_{L_1(w)} = \sum_{k\geq 1} |x(k)| w(k)^{-1} < \infty \right\}.$$

For easier notation, we denote the sequence $g_\theta = (\theta^k)_{k\geq 1}$. WLOG we assume that $d \geq 2$ and $p \in (0,1)$ since otherwise the system goes back to JSQ(d) (refer to Graham [39] for their results). Notice that by induction we can show that for $k \geq 2$,

$$\lambda^k (1-p)^{k-1} < s^I_k < \lambda^k$$

$$\lambda^{k-1}(1-p)^{k-1} < \pi_k < \lambda^{k-1}$$

which means that both s^I and π have exponential decay. In the rest of the chapter, we assume that w satisfies the following condition,

$$\exists c, d > 0, \forall k \geq 1, 0 < cw(k+1) \leq w(k) \leq dw(k+1). \tag{4.34}$$

This condition implies $w(1)d(1/d)^k \leq w(k) \leq w(1)c(1/c)^k$, which means w is bounded by geometric sequences.

Theorem 19 (Functional Central Limit Theorem in Equilibrium)**.** *Let w satisfies condition (4.34), then*

1. *In $L_2(w)$, the operator $\mathcal{K}$ is bounded, and Equation (4.33) has a unique solution $d_t = e^{\mathcal{K}t}d(0)$. The assumptions and conclusions hold for $w = \pi$ and $w = g_\theta$ for $\theta > 0$.*

2. *In addition, let w be such that s^I is in $L_1(w)$. The SDE (4.32) has a unique solution*
$$D(t) = e^{\mathcal{K}t}D(0) + \int_0^t e^{\mathcal{K}(t-s)}dB(s)$$
in $L_2(w)$. This is the case for $w = g_\theta$ for $\theta \geq \lambda$ when $d \geq 2$.

Proof. Using the condition in Equation (4.34) and our convexity bounds, we have

$$\begin{aligned}
&\|\mathcal{K}x\|_{L_2(w)} \\
= \; & \sum_{k\geq 1}\Big[\Big(\lambda(1-p)+\lambda pd\left(s^I_{k-1}\right)^{d-1}\Big)x_{k-1} \\
&\qquad -\Big(\lambda(1-p)+\lambda pd\left(s^I_k\right)^{d-1}+1\Big)x_k + x_{k+1}\Big]^2 w(k)^{-1} \\
\leq \; & 3\Bigg(\sum_{k\geq 1}\Big(\lambda(1-p)+\lambda pd\left(s^I_{k-1}\right)^{d-1}\Big)^2 x^2_{k-1}w(k)^{-1} \\
&\qquad +\Big(\lambda(1-p)+\lambda pd\left(s^I_k\right)^{d-1}+1\Big)^2 x_k^2 w(k)^{-1} + x^2_{k+1}w(k)^{-1}\Bigg) \\
\leq \; & 3\Bigg(\sum_{k\geq 1}(\lambda(1-p)+\lambda pd)^2 x^2_{k-1} d w(k-1)^{-1} \\
&\qquad +(\lambda(1-p)+\lambda pd+1)^2 x_k^2 w(k)^{-1} + x^2_{k+1}c^{-1}w(k+1)^{-1}\Bigg) \\
\leq \; & 3\left(d(\lambda(1-p)+\lambda pd)^2+(\lambda(1-p)+\lambda pd+1)^2+c^{-1}\right)\|x\|_{L_2(w)}. \qquad (4.35)
\end{aligned}$$

Then by applying Gronwall's lemma we have the uniqueness result. When B is an Hilbertian Brownian motion, the formula for $D(t)$ yields a well-defined solution. □

4.3.3 Interchanging Limits

Our goal in this section is to prove the following diagram commutes.

$$\begin{tikzcd}
D^N(t) \arrow[r, "N\to\infty"] \arrow[d, "t\to\infty"'] & D(t) \arrow[d, "t\to\infty"] \\
D^N(\infty) \arrow[r, "N\to\infty"'] & D(\infty)
\end{tikzcd}$$

We have showed in Section 4.3.1 that $D^N(t) \overset{d}{\to} D(t)$, and the existence and uniqueness of $D(t)$. Now we will show the existence and uniqueness of the equilibrium point $D(\infty)$ of the diffusion limit, and show the weak convergence of invariant measure $D^N(\infty)$ to $D(\infty)$. The proof idea of the interchanging limits of diffusion limits takes the following list of steps:

1. Prove the equilibrium operator $\mathcal{K}$ has bounded spectral gap in the self-adjoint space $L_2(\pi)$, which implies exponential stability of linearized solution d_t in $L_2(\pi)$. (Theorem 20)
2. Prove exponential stability of fluid limit $s(t)$ in non self-adjoint space $L_2(g(\theta))$, by constructing a specific birth-death process and obtain exponential stability of its solution $z(t)$ via step 1, then bounding the difference between the fluid limit $s(t)$ and $z(t)$. (Theorem 21, Lemma 12, Lemma 13)
3. Show the infinite horizon bound in space $L_2(g(\theta))$ using the exponential stability result of $s(t)$ in step 2. (Theorem 22)
4. Show the weak convergence of stationary distributions $D^N(\infty)$ to the equilibrium point $D(\infty)$ of the diffusion limit $D(t)$. (Theorem 23).

Consider $\mathcal{A} = \mathcal{K}^*$, the infinitesimal generator of the sub-Markovian birth death process with birth rates $\lambda_k = \lambda(1-p) + \lambda p d (s_k^I)^{d-1}$ and death rates $\mu_k = 1$

for $k \geq 1$. Let $Q(x) = (Q_n(x))_{n\geq 1}$ denote an eigenvector for $\mathcal{A}$ of eigenvalue $-x$. Then, we have $\lambda_1 Q_2(x) = (\lambda_1 + \mu_1 - x)Q_1(x)$ and $\lambda_n Q_{n+1}(x) = (\lambda_n + \mu_n - x)Q_n(x) - \mu_n Q_{n-1}(x)$ for $n \geq 2$. Such a sequence of polynomials is orthogonal with respect to a probability measure ψ on $\mathbb{R}^+$ such that

$$\text{diag}(\pi^{-1}) = \int_0^\infty Q(x)Q(x)^* \psi(dx).$$

Such a probability measure is called the spectral measure, with its support S called the spectrum. We denote the spectral gap $\gamma = \min S$. The representation formula of Karlin and McGregor [47, 46] yields

$$e^{\mathcal{K}t} = \text{diag}(\pi) \int_0^\infty e^{-xt} Q(x)Q(x)^* \psi(dx). \tag{4.36}$$

Therefore, we have the following lemma which gives the solution of the unique equilibrium point of the OU process.

Lemma 11. *The OU process $D(t)$ in Theorem 19, its equilibrium point $D(\infty)$, and its covariance matrix $\Sigma(\infty)$ can be written as*

$$D(t) = diag(\pi) \int_S e^{-xt} Q(x)^* \left(D(0) + \int_0^t e^{xs} dB(s) \right) Q(x)\psi(dx)$$

$$D(\infty) \;\;=\;\; diag(\pi) \int_S \left(Q(x)^* \int_0^\infty e^{-xt} dB(t) \right) Q(x)\psi(dx)$$

$$\Sigma(\infty) \;\;=\;\; diag(\pi) \int_{S^2} \frac{Q(x)^* diag(v) Q(y)}{x+y} Q(x)Q(y)^* \psi(dx)\psi(dy) diag(\pi) \tag{4.37}$$

Proof. We have the unique solution $D(t)$ as

$$\begin{aligned}
& D(t) \\
= \; & e^{\mathcal{K}t}D(0) + \int_0^t e^{\mathcal{K}(t-s)}dB(s) \\
= \; & \text{diag}(\pi)\int_0^\infty e^{-xt}Q(x)Q(x)^*\psi(dx)D(0) \\
& \quad + \int_0^t \left(\text{diag}(\pi)\int_0^\infty e^{-x(t-s)}Q(x)Q(x)^*\psi(dx)\right)dB(s) \\
= \; & \text{diag}(\pi)\int_0^\infty e^{-xt}Q(x)^*D(0)Q(x)\psi(dx) \\
& \quad + \text{diag}(\pi)\int_0^\infty e^{-xt}Q(x)^*\left(\int_0^t e^{xs}dB(s)\right)Q(x)\psi(dx) \\
= \; & \text{diag}(\pi)\int_S e^{-xt}Q(x)^*\left(D(0)+\int_0^t e^{xs}dB(s)\right)Q(x)\psi(dx).
\end{aligned}$$

Note that here we define $Q(x) = (Q_1(x), Q_2(x), \cdots, Q_n(x), \cdots)^\top$, which is a infinite dimensional column vector of polynomials. Thus $Q(x)^*D(0)$ and $Q(x)^*dB(s)$ are 1-dimensional numbers and are exchangeable with $Q(x)$ in matrix multiplication.

For the equilibrium point $D(\infty)$ of the OU process, we have

$$\begin{aligned}
D(\infty) &= \int_0^\infty e^{\mathcal{K}t}dB(t) \\
&= \int_0^\infty \left(\text{diag}(\pi)\int_0^\infty e^{-xt}Q(x)Q(x)^*\psi(dx)\right)dB(t) \\
&= \text{diag}(\pi)\int_S \left(Q(x)^*\int_0^\infty e^{-xt}dB(t)\right)Q(x)\psi(dx),
\end{aligned}$$

and its covariance matrix $\Sigma(\infty)$ is as follows,

$$\begin{aligned}
& \Sigma(\infty) \\
= \;& \int_0^\infty e^{\mathcal{K}t}\mathbb{E}[B(1), B(1)^*]e^{\mathcal{K}^*t}dt \\
= \;& \int_0^\infty e^{\mathcal{K}t}\mathbf{diag}(v)e^{\mathcal{K}^*t}dt \\
= \;& \int_0^\infty \Big[\int_S \big(\mathbf{diag}(\pi)e^{-xt}Q(x)Q(x)^*\psi(dx)\big) \\
& \qquad\qquad \cdot \mathbf{diag}(v)\int_S \big(\mathbf{diag}(\pi)e^{-yt}Q(y)Q(y)^*\psi(dy)\big)\Big]\,dt \\
= \;& \mathbf{diag}(\pi) \\
& \quad \cdot \int_{S^2}\left(\int_0^\infty e^{-(x+y)t}dt\right)Q(x)(Q(x)^*\mathbf{diag}(v)Q(y))Q(y)^*\psi(dx)\psi(dy)\mathbf{diag}(\pi) \\
= \;& \mathbf{diag}(\pi)\int_{S^2}\frac{Q(x)^*\mathbf{diag}(v)Q(y)}{x+y}Q(x)Q(y)^*\psi(dx)\psi(dy)\mathbf{diag}(\pi).
\end{aligned}$$

□

Theorem 20 (Spectral Gap for self-adjoint case)**.** *The operator $\mathcal{K}$ is bounded self-joint in $L_2(\pi)$. The least point γ of the spectrum of $\mathcal{K}$ is such that $0 < \gamma \leq (\sqrt{\lambda(1-p)} - 1)^2$. The solution $d(t) = e^{\mathcal{K}t}d(0)$ for Equation (4.33) in $L_2(\pi)$ satisfies $\|d(t)\|_{L_2(\pi)} \leq e^{-\gamma t}\|d(0)\|_{L_2(\pi)}$.*

Proof. The potential coefficients π solve the detailed balance equations for $\mathcal{A}$ and hence $\mathcal{K} = \mathcal{A}^*$ is self-adjoint in $L_2(\pi)$. It is established in Theorem 5.1 and Theorem 5.3 in Doorn [27] that $\gamma > 0$ if and only if

$$\sigma = \left(\sqrt{\lim_k \lambda_k} - \sqrt{\lim_k \mu_k}\right)^2 = \left(\sqrt{\lambda(1-p)} - 1\right)^2 > 0.$$

For exponential stability, we have $\|d(t)\|^2_{L_2(\pi)} = \big(e^{\mathcal{K}t}d(0), e^{\mathcal{K}t}d(0)\big)_{L_2(\pi)}$ and the fact that $e^{\mathcal{K}t}$ is self-adjoint in $L_2(\pi)$ and the spectral representation yield

$$\begin{aligned}
\left(e^{\mathcal{K}t}d(0), e^{\mathcal{K}t}d(0)\right)_{L_2(\pi)} &= \left(d(0), e^{2\mathcal{K}t}d(0)\right)_{L_2(\pi)} \\
&= \int_S e^{-2xt} d(0)^* Q(x) Q(x)^* d(0) \psi(dx) \\
&\leq e^{-2\gamma t} \int_S d(0)^* Q(x) Q(x)^* d(0) \psi(dx) \\
&= e^{-2\gamma t} (d(0), d(0))_{L_2(\pi)}.
\end{aligned}$$

□

For the proof of exponential stability for non self-adjoint case, we modify an argument of Graham [39]. We first consider the centered dynamical system $y(t) = s(t) - s^I$, then y solves the centered equation

$$\dot{y}(t) = F(s^I + y) = \mathcal{K}y(t) + H(s^I, y(t)),$$

or

$$\begin{aligned}
\dot{y}_k(t) &= [\lambda(1-p) + \lambda p d (s^I_{k-1})^{d-1}] y_{k-1}(t) + \lambda p B\left(s^I_{k-1}, y_{k-1}(t)\right) \\
&\quad -[\lambda(1-p) + \lambda p d (s^I_k)^{d-1} + 1] y_k(t) - \lambda p B\left(s^I_k, y_k(t)\right) + y_{k+1}(t) \quad (4.38)
\end{aligned}$$

We also have

$$\dot{y}_k(t) + \dot{y}_{k+1}(t) + \cdots = [\lambda(1-p) + \lambda p d (s^I_{k-1})^{d-1}] y_{k-1}(t) + \lambda p B\left(s^I_{k-1}, y_{k-1}(t)\right) - y_k(t).$$

Lemma 12. *Let $\hat{A}$ be the generator of the sub-Markovian birth and death process with birth rate $\hat{\lambda}_k \geq 0$ and death rate 1 for $k \geq 1$. Assume* $\sup_k \hat{\lambda}_k < \infty$. *Let $z(t)$ solves $\dot{z} = \hat{A}^* z$ in ℓ^0_1. Let $h(t)$ be given in ℓ^0_1 by*

$$h_k(t) = \sum_{i \geq k} (z_i(t) - y_i(t)), \quad k \geq 1$$

Then,

(1) *Let* $\hat{\lambda}_k \geq [\lambda(1-p) + \lambda pd(s_k^I)^{d-1}] + \lambda p(1 + (2^d - d - 2)s_k^I)$ *for* $k \geq 1$, $y(0) \geq 0$ *and* $h(0) \geq 0$. *Then* $h(t) \geq 0$ *for* $t \geq 0$.

(2) *Let* $\hat{\lambda}_k \geq [\lambda(1-p) + \lambda pd(s_k^I)^{d-1}]$ *for* $k \geq 1$, $y(0) \leq 0$ *and* $h(0) \leq 0$. *Then* $h(t) \leq 0$ *for* $t \geq 0$.

Proof. We first prove (1). We can assume WLOG that $\hat{\lambda}_k > [\lambda(1-p) + \lambda pd(s_k^I)^{d-1}] + \lambda p(1 + (2^d - d - 2)s_k^I)$ for $k \geq 1$. Since $z(t) = e^{\hat{A}^* t} z(0)$ depends continuously on $z(0)$ in ℓ_1^0, we may assume $h(0) > 0$. Let $\tau = \inf\{t \geq 0 : \{k \geq 1 : h_k(t) = 0\} = \emptyset\}$ be the first time when $h_k = 0$ for some $k \geq 1$. We know that $\tau > 0$ and the result holds when $\tau = \infty$.

If $\tau < \infty$, we have

$$\begin{aligned}\dot{h}_k(\tau) &= \hat{\lambda}_{k-1} y_{k-1}(\tau) - [\lambda(1-p) + \lambda pd(s_{k-1}^I)^{d-1}] y_{k-1}(\tau) - \lambda p B\left(s_{k-1}^I, y_{k-1}(\tau)\right) \\ &\quad + \hat{\lambda}_{k-1}(z_{k-1}(\tau) - y_{k-1}(\tau)) - (z_k(\tau) - y_k(\tau)).\end{aligned}$$

Lemma 5 and $y(0) \geq 0$ implies that $y(t) \geq 0$ for all $t \geq 0$. Any by Lemma 9 we have that

$$B\left(s_{k-1}^I, y_{k-1}(\tau)\right) \leq y_{k-1}^d + (2^d - d - 2)s_{k-1}^I y_{k-1}^2 \leq \left(1 + (2^d - d - 2)s_{k-1}^I\right) y_{k-1}.$$

Therefore by the assumption that $\hat{\lambda}_k \geq [\lambda(1-p) + \lambda pd(s_k^I)^{d-1}] + \lambda p(1 + (2^d - d - 2)s_k^I)$ we have that

$$\hat{\lambda}_{k-1} y_{k-1}(\tau) - [\lambda(1-p) + \lambda pd(s_{k-1}^I)^{d-1}] y_{k-1}(\tau) - \lambda p B\left(s_{k-1}^I, y_{k-1}(\tau)\right) \geq 0,$$

and equality holds only when $y_{k-1} = 0$. For $k \in \mathcal{Z} = \{k \geq 1 : h_k(\tau) = 0\}$ we have

$$z_{k-1}(\tau) - y_{k-1}(\tau) = h_{k-1}(\tau) - h_k(\tau) = h_{k-1}(\tau) \geq 0,$$

$$z_k(\tau) - y_k(\tau) = h_k(\tau) - h_{k+1}(\tau) = -h_{k+1}(\tau) \leq 0,$$

hence $\dot{h}_k(\tau) \geq 0$ with equality only when $k-1 \in \mathcal{Z} \cup \{0\}$ and $k+1 \in \mathcal{Z}$. We also know that $h_k(t) > 0$ for $t < \tau$ and $h_k(\tau) = 0$ which implies $\dot{h}_k(\tau) \leq 0$. Thus $\dot{h}_k(\tau) = 0$ and $z_{k-1}(\tau) = y_{k-1}(\tau) = 0$ and $k-1 \in \mathcal{Z} \cup \{0\}$ and $k+1 \in \mathcal{Z}$. By induction we have that $z_k(\tau) = y_k(\tau) = 0$ for all $k \geq 1$, which means $z(t) = y(t) = 0$ for all $t \geq \tau$, thus $h(t) \geq 0$ for $t \geq 0$. The proof for (2) follows similarly. □

Lemma 13. *For any $0 \leq \theta < 1$ there exists $K_\theta < \infty$ such that for x in $L_2(g_\theta) \subset \ell_1^0$*

$$\|(x_k + x_{k+1} + \cdots)_{k \geq 1}\|_{L_2(g_\theta)} \leq K_\theta \|x\|_{L_2(g_\theta)}.$$

Proof.

$$\begin{aligned} & \|(x_k + x_{k+1} + \cdots)_{k\geq 1}\|_{L_2(g_\theta)} \\ = \ & \sum_{k \geq 1} (x_k + x_{k+1} + \cdots)^2 \theta^{-k} \\ \leq \ & \sum_{k\geq 1} n \left(x_k^2 + x_{k+1}^2 + \cdots + x_{k+n-2}^2 + (x_{k+n-1} + x_{k+n} + \cdots)^2 \right) \theta^{-k} \\ \leq \ & n(1 + \theta + \cdots + \theta^{n-2}) \sum_{k\geq 1} x_k^2 \theta^{-k} + n\theta^{n-1} \sum_{k \geq 1} (x_k + x_{k+1} + \cdots)^2 \theta^{-k}. \end{aligned}$$

Since this holds for any $n \geq 1$ we can choose n large enough such that $n\theta^{n-1} < 1$, then

$$(1 - n\theta^{n-1}) \|(x_k + x_{k+1} + \cdots)_{k\geq 1}\|_{L_2(g_\theta)} \leq n(1 + \theta + \cdots + \theta^{n-2}) \|x\|_{L_2(g_\theta)}.$$

Letting $K_\theta = n(1 + \theta + \cdots + \theta^{n-2})/(1 - n\theta^{n-1})$, we have the result. □

Now we finish the proof of Theorem 21 using the previous two lemmas.

Theorem 21 (Exponential stability for non self-adjoint case)**.** *Let $\lambda \leq \theta < 1$ and s be the solution to Equation (4.5) starting at $s(0)$ in $\mathcal{S} \cap L_2(g_\theta)$. There exists $\gamma_\theta > 0$ and $C_\theta < \infty$ such that*

$$\|s(t) - s^I\|_{L_2(g_\theta)} \leq e^{-\gamma_\theta t} C_\theta \|s(0) - s^I\|_{L_2(g_\theta)}.$$

Proof. Assume $s(0) \in \mathcal{S}$ is in $L_2(g_\theta)$. Denote $s^+(0) = \max\{s(0), s^I\}$ and $s^-(0) = \min\{s(0), s^I\}$, and s^+, s^- as the corresponding solution to (4.5) with such initial condition. Then by Lemma 5 we have that $s^+(t) \leq s(t) \leq s^-(t)$ and $s^+(t) \leq s^I \leq s^-(t)$ for all $t \geq 0$. Again we use $y(t) = s(t) - s^I$ to denote the solution to the recentered equation, and that $y^+(t) = s^+(t) - s^I, y^-(t) = s^-(t) - s^I$. We also have

$$|y(0)| = \max\{y(0)^+, y(0)^-\}, |y(t)| \leq \max\{y(t)^+, -y(t)^-\}, t \geq 0.$$

Now consider a birth death process with generator $\hat{\mathcal{A}}$ where birth rate $\hat{\lambda}_k$ is as follows,

$$\hat{\lambda}_k = \max\{[\lambda(1-p) + \lambda p d (s_k^I)^{d-1}] + \lambda p(1 + (2^d - d - 2)s_k^I), \theta\}, k \geq 1.$$

For $\lambda \leq \theta < 1$, we know that for large enough k $\hat{\lambda}_k$ is equal to θ. Using the same method as in the proof of theorem 20, we have that the spectral gap $\hat{\gamma}$ for the birth death process with generator $\hat{\mathcal{A}}$ satisfies that $0 < \hat{\gamma} \leq \hat{\sigma} = (\sqrt{\theta} - 1)^2$. This means that the solution $z(t)$ to $\dot{z} = \hat{\mathcal{A}}^* z$ has exponential stability, i.e.

$$\|z(t)\|_{L_2(\hat{\pi})} \leq e^{-\hat{\gamma} t} \|z(0)\|_{L_2(\hat{\pi})}, t \geq 0$$

where $\hat{\pi}$ is the stationary distribution to $\hat{\mathcal{A}}^*$.

We know that

$$\hat{\pi}_k = \prod_{i=1}^{k-1} \hat{\lambda}_k = \theta^{k-1} \prod_{i=1}^{k-1} \max\{\theta^{-1}[\lambda + \lambda p d (s_k^I)^{d-1} + \lambda p (2^d - d - 2) s_k^I], 1\} \geq \theta^{k-1}$$

and the product converges. Thus $\hat{\pi}_k = O(\theta^k)$ and $\theta^k = O(\hat{\pi}_k)$ and therefore the two norms $L_2(\hat{\pi})$ and $L_2(g_\theta)$ are equivalent, so there exists $c, d > 0$ such that

$$\|z(t)\|_{L_2(g_\theta)} \leq d\|z(t)\|_{L_2(\hat{\pi})} \leq d e^{-\hat{\gamma} t} \|z(0)\|_{L_2(\hat{\pi})} \leq c d e^{-\hat{\gamma} t} \|z(0)\|_{L_2(g_\theta)}.$$

Let z^+, z^- be the corresponding solutions to $z^+ = \hat{\mathcal{A}}^* z^+$ and $z^- = \hat{\mathcal{A}}^* z^-$ starting $y(0)^+ \geq 0$ and $y(0)^- \leq 0$ respectively. Then, by Lemma 12 and Lemma 13, we have

$$\begin{aligned}\|y^+(t)\|_{L_2(g_\theta)} &\leq \|(y_k^+(t)+y_{k+1}^+(t)+\cdots)_{k\geq 1}\|_{L_2(g_\theta)}\\ &\leq \|(z_k^+(t)+z_{k+1}^+(t)+\cdots)_{k\geq 1}\|_{L_2(g_\theta)}\\ &\leq K_\theta\|z^+(t)\|_{L_2(g_\theta)}\\ &\leq cdK_\theta e^{-\hat{\gamma}t}\|y^+(0)\|_{L_2(g_\theta)},\end{aligned}$$

and similarly $\|y^-(t)\|_{L_2(g_\theta)} \leq cdK_\theta e^{-\hat{\gamma}t}\|y^-(0)\|_{L_2(g_\theta)}$. Letting $\gamma_\theta = \hat{\gamma}$ and $C_\theta = c^2d^2K_\theta^2$, we have

$$\begin{aligned}\|y(t)\|_{L_2(g_\theta)}^2 &\leq \|y^+(t)\|_{L_2(g_\theta)}^2+\|y^-(t)\|_{L_2(g_\theta)}^2 \\ &\leq e^{-2\gamma_\theta t}C_\theta\left(\|y^+(0)\|_{L_2(g_\theta)}^2+\|y^-(0)\|_{L_2(g_\theta)}^2\right)\\ &= e^{-2\gamma_\theta t}C_\theta\|y(0)\|_{L_2(g_\theta)}^2.\end{aligned} \tag{4.39}$$

□

Theorem 22 (Infinite Horizon Bound). *Assume $\lambda \leq \theta < 1$, then*

$$\limsup_{N\to\infty}\mathbb{E}\left(\|D^N(0)\|_{L_2(g_\theta)}^2\right) < \infty \Rightarrow \limsup_{N\to\infty}\sup_{t\geq 0}\mathbb{E}\left(\|D^N(t)\|_{L_2(g_\theta)}^2\right)$$

Proof. We consider the case when $s(0) = s^I$. Since s^I is the equilibrium, we have $s(t) = s^I$ for all $t \geq 0$. Let $s(\nu, h)$ be the solution of Equation (4.5) at time $h \geq 0$ with initial value ν. For $t_0 \geq 0$ let $D^N(t_0, h) = \sqrt{N}(S^N(t_0+h) - s(S^N(t_0), h))$. Then we have $D^N(t_0+h) = D^N(t_0, h) + \sqrt{N}\left(s(S^N(t_0), h) - s^I\right)$. By Lemma 21,

$$\|D^N(t_0+h)\|_{L_2(g_\theta)} \leq \|D^N(t_0,h)\|_2 + C_\theta e^{-\gamma_\theta h}\|D^N(t_0)\|_{L_2(g_\theta)}. \tag{4.40}$$

The conditional distribution of $D^N(t_0, h)$ given $S^N(t_0) = s$ is the distribution of D^N started with $S^N(t_0) = s(0) = s$. In particular, $D^N(t_0) = D^N(t_0, 0) = 0$.

Following a similar argument as in Equation (4.31), we have that there exists constant $C_T > 0$ such that

$$\begin{aligned}\sup_{0\le h\le T} \|D^N(t_0,h)\|_{L_2(g_\theta)} \quad &\le \quad C_T\left(\frac{1}{\sqrt{N}}\|s^I\|_{L_2(g_\theta)} + \frac{1}{N}C_\theta\|D^N(t_0)\|_{L_2(g_\theta)}\right.\\ &\qquad \left. + \sup_{0\le h\le T}\left\|M^N(t_0+h) - M^N(t_0)\right\|_{L_2(g_\theta)}\right).\end{aligned}$$

Combined with (4.40), we have that for some $L_T > 0$ and $0 \le h \le T$,

$$\mathbb{E}\left(\|D^N(t_0+h)\|^2_{L_2(g_\theta)}\right) \le L_T + 2\left(\frac{C_T}{N} + e^{-\gamma_\theta h}\right)^2 C_\theta^2 \mathbb{E}\left(\|D^N(t_0)\|^2_{L_2(g_\theta)}\right). \quad (4.41)$$

Now for fixed T large enough we have $8e^{-2\gamma_\theta T}C_\theta^2 \le \epsilon < 1$. Then uniformly for $N \ge C_T e^{\gamma_\theta T}$, for $m \in \mathbb{N}$, we have

$$\mathbb{E}\left(\|D^N((m+1)T)\|^2_{L_2(g_\theta)}\right) \le L_T + \epsilon\mathbb{E}\left(\|D^N(mT)\|^2_{L_2(g_\theta)}\right).$$

By induction,

$$\begin{aligned}\mathbb{E}\left(\|D^N(mT)\|^2_{L_2(g_\theta)}\right) \quad &\le \quad L_T\sum_{j=1}^{m}\epsilon^{j-1} + \epsilon^m\mathbb{E}\left(\|D^N(0)\|^2_{L_2(g_\theta)}\right)\\ &\le \quad \frac{L_T}{1-\epsilon} + \mathbb{E}\left(\|D^N(0)\|^2_{L_2(g_\theta)}\right).\end{aligned}$$

From (4.41), we know that

$$\sup_{0\le h\le T}\mathbb{E}\left(\|D^N(mT+h)\|^2_{L_2(g_\theta)}\right) \le L_T + 8C_\theta^2\mathbb{E}\left(\|D^N(mT)\|^2_{L_2(g_\theta)}\right),$$

hence we have the infinite horizon bound

$$\sup_{t\ge 0}\mathbb{E}\left(\|D^N(t)\|^2_{L_2(g_\theta)}\right) \le L_T + 8C_\theta^2\mathbb{E}\left(\frac{L_T}{1-\epsilon} + \mathbb{E}\|D^N(0)\|^2_{L_2(g_\theta)}\right).$$

Ergodicity and the Fatou Lemma yield that for $D^N(\infty)$

$$\mathbb{E}(\|D^N(\infty)\|^2_{L_2(g_\theta)}) \le \liminf_{t\ge 0}\mathbb{E}(\|D^N(t)\|^2_{L_2(g_\theta)}) \le \sup_{t\ge 0}\mathbb{E}(\|D^N(t)\|^2_{L_2(g_\theta)}) < \infty.$$

□

We now prove that the interchanging of limits is valid, through the following steps:

1) The sequence $(D^N(\infty), N \geq 1)$ is tight.
2) There is a unique possible limit to any convergent subsequence of $(D^N(\infty), N \geq 1)$.

Theorem 23. *The stationary distribution $D^N(\infty)$ of the diffusion process $D^N(t)$ converges weakly to the equilibrium point of the diffusion limit $D(\infty)$, whose explicit form is specified in Equation (4.37).*

Proof. Since for any $K > 0$, using Markov inequality we have

$$\begin{aligned} P(\|D^N(\infty)\|_{L_2(g_\theta)} > K) &= \lim_{t\to\infty} P(\|D^N(t)\|_{L_2(g_\theta)} > K) \\ &= \lim_{t\to\infty} \frac{\mathbb{E}[\|D^N(t)\|^2_{L_2(g_\theta)}]}{K^2} \\ &= O\left(\frac{1}{K^2}\right). \end{aligned}$$

This shows that $(D^N(\infty), N \geq 1)$ is tight. Now we only need to prove that there is a unique possible limit to any convergent subsequence of $(D^N(\infty), N \geq 1)$. We still denote by $D^N(\infty)$ such a converging subsequence. Its limit is denoted by ν. By properties of Markov processes, $D^N(t)$ with initial condition $D^N(0) = D^N(\infty)$ is a stationary process, hence $D(t) = \nu$ for any t. Then $D(\infty) = \nu$, which proved that any convergent subsequence of $D^N(\infty)$ converge to $D(\infty)$. □

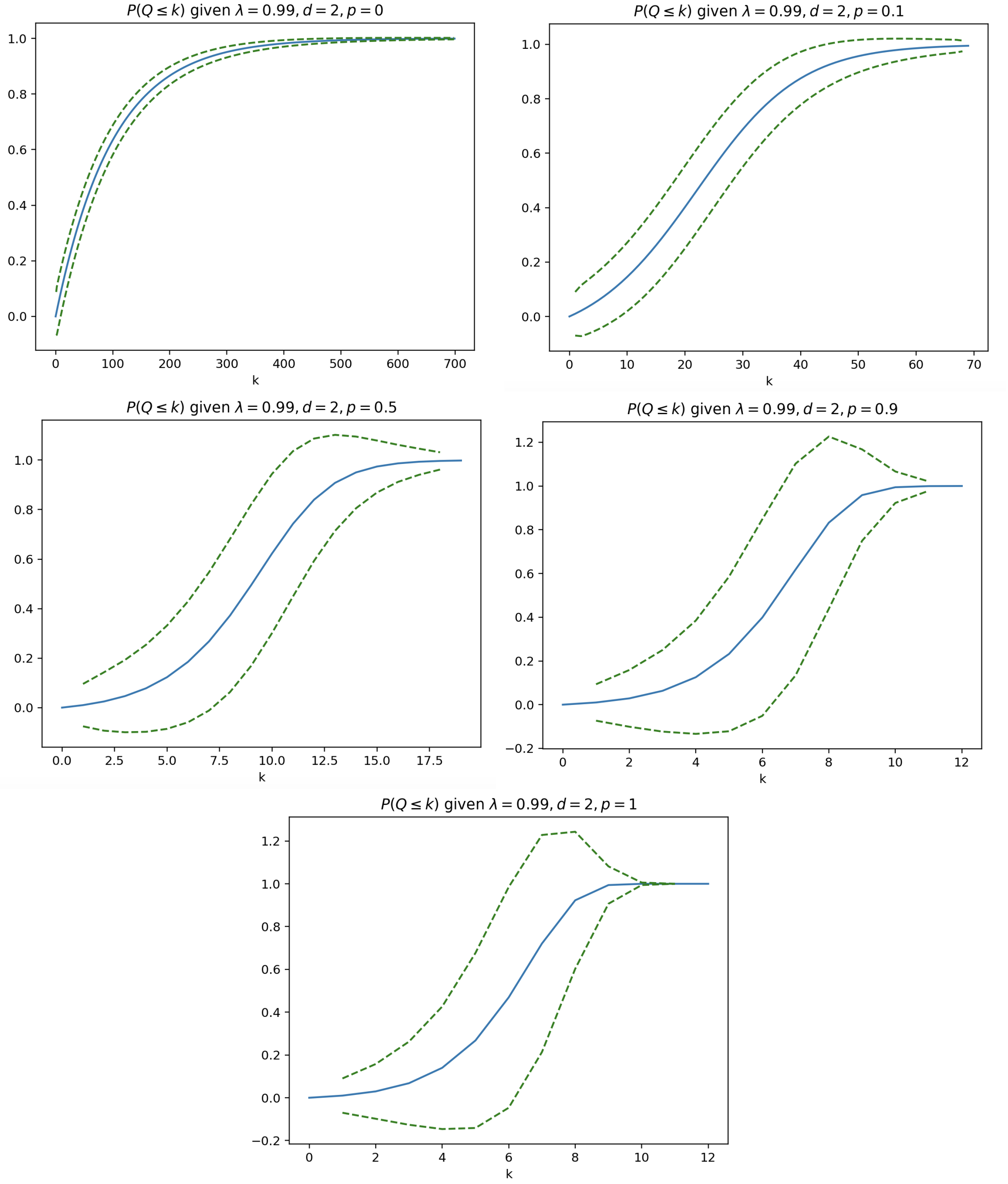


Figure 4.3: $\mathbb{P}(Q \leq k)$ for various values of p

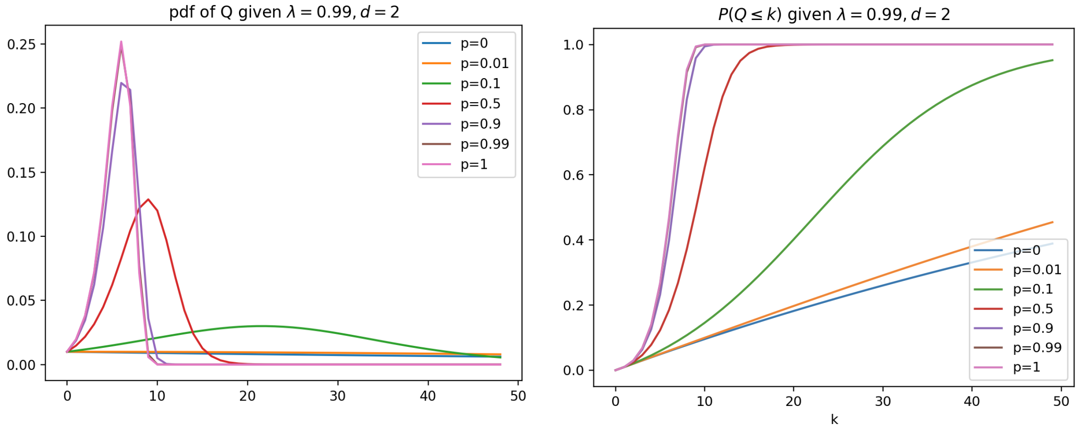


Figure 4.4: The pdf and cdf of Q for various values of p

Now that we have proved both fluid and diffusion limits for the queue length process, we can apply those results to some numerical examples. In Figure 4.3, we provide five plots where the flexibility parameter p changes throughout each plot. The green dotted lines indicate one standard deviation computed according to Theorem 18 with a cutoff at 1,000 iterations. We observe that p has a large effect on the shape of the distribution. In fact, by increasing p, the distribution develops an inflection point. Moreover, we observe that by having ten percent of flexible customers reduces the max queue length by an order of 10. As one continues to increase p, the max queue length decreases, but not as much as the initial few flexible customers.

In Figure 4.4, we plot the probability density function and the cdf of the queue length for a variety of values of p. On the left of Figure 4.4, we observe that as we increase p, the pdf mode moves to the left. Moreover, as p decreases, the pdf becomes more flat. On the right of Figure 4.4, we see that flat behavior of systems with small p is confirmed since the cdf of the queue length appears to have a linear shape.

4.4 Insights on Dependence on p and d

In this section, we provide new insights about our model with flexible customers. To this end, we prove two new results and also provide numerical experiments that validate our fluid and diffusion approximations. The first result shows that we can obtain a closed form solution for the tail cdf of the queue length distribution. The second result proves upper and lower bounds on the mean, second moment, and variance of the queue length process. We first start with a closed form solution of the steady state tail cdf.

4.4.1 Steady State Fluid Limit Solution

The steady state of the fluid limit admits a unique closed-form solution for the tail cdf. In order to show this result, we exploit a similar argument used by Rabinovich et al. [73].

Proposition 7 (Closed-form Solution of the Steady State)**.** *The steady state solution of the queueing model s^I satisfies a nonlinear recursion*

$$s_i^I = \lambda(1-p)s_{i-1}^I + \lambda p (s_{i-1}^I)^d \quad \textit{for all } i \geq 2.$$

$$\lambda(1-p)\left(s_{i-1}^I - s_i^I\right) + \lambda p \left(\left(s_{i-1}^I\right)^d - \left(s_i^I\right)^d\right) - \left(s_i^I - s_{i+1}^I\right) = 0$$

and has a unique closed-form solution given by

$$\begin{aligned} s_i^I \; &= \; \sum_{k_1=1}^{d} \sum_{k_2=k_1}^{dk_1} \sum_{k_3=k_2}^{dk_2} \cdots \sum_{k_i=k_{i-1}}^{dk_{i-1}} \binom{1}{\frac{k_1-1}{d-1}} \binom{k_1}{\frac{k_2-k_1}{d-1}} \cdots \binom{k_{i-1}}{\frac{k_i-k_{i-1}}{d-1}} \\ & \qquad \cdot \left(\lambda(1-p)\right)^{1+k_1+k_2+\cdots+k_{i-1}-\frac{k_i-1}{d-1}} \left(\lambda p\right)^{\frac{k_i-1}{d-1}} . \end{aligned}$$

Proof. By Rabinovich et al. [73], the solution to any first-order recursion equation given by

$$s_{i+1} = P\left(s_i\right)$$

can be written as

$$s_i = \langle e|T^i|s\rangle.$$

Here $|s\rangle = \left\{s_0^j\right\}_{j=0}^{\infty}$ and $\langle e| = \left[\delta_{j1}\right]_{j=0}^{\infty}$ where δ_{jk} is the Kronecker symbol. T is a transfer matrix that transforms the column $\left\{s_i^j\right\}$ to a column $\left\{\left[P\left(s_i\right)\right]^j\right\}$.

In our case,

$$P\left(s_i^I\right) = \lambda(1-p)s_i^I + \lambda p\left(s_i^I\right)^d$$

and $\left\{\left[P\left(s_i^I\right)\right]^j\right\}$ can be expanded as the following:

$$\begin{aligned}\left[\lambda(1-p)s_i^I + \lambda p\left(s_i^I\right)^d\right]^j &= \sum_{l=0}^{j}\binom{j}{l}(\lambda p)^l\left(\left(s_i^I\right)^d\right)^l(\lambda(1-p))^{j-l}\left(s_i^I\right)^{j-l} \\ &= \sum_{l=0}^{j}\binom{j}{l}(\lambda p)^l(\lambda(1-p))^{j-l}\left(s_i^I\right)^{j+(d-1)l}.\end{aligned}$$

Denoting $k = j + (d-1)l$ so that $l = \frac{k-j}{d-1}$, we have

$$\sum_{k=j}^{dj}\binom{j}{\frac{k-j}{d-1}}(\lambda p)^{\frac{k-j}{d-1}}(\lambda(1-p))^{j-\frac{k-j}{d-1}}\left(s_i^I\right)^k.$$

Thus the matrix elements T_{jk} are

$$T_{jk} = \binom{j}{\frac{k-j}{d-1}}(\lambda p)^{\frac{k-j}{d-1}}(\lambda(1-p))^{j-\frac{k-j}{d-1}}.$$

Given $s_0^I = 1$, the solution to our nonlinear recursion $s_i^I = \langle e|T^i|s^I\rangle$ is the sum of all elements in the first row of T^i:

$$
\begin{aligned}
s_i^I &= \sum_{k_i=0}^{d^i} \left(T^i\right)_{1,k_i} \\
&= \sum_{k_i=0}^{d^i} \sum_{k_{i-1}=0}^{d^i} \cdots \sum_{k_1=0}^{d^i} T_{1,k_1} T_{k_1,k_2} \cdots T_{k_{i-1},k_i} \\
&= \sum_{k_i=0}^{d^i} \sum_{k_{i-1}=0}^{d^i} \cdots \sum_{k_1=0}^{d^i} \binom{1}{\frac{k_1-1}{d-1}} \binom{k_1}{\frac{k_2-k_1}{d-1}} \cdots \binom{k_{i-1}}{\frac{k_i-k_{i-1}}{d-1}} \\
&\qquad\qquad \cdot \left(\lambda(1-p)\right)^{1+k_1+\cdots+k_{i-1}-\frac{k_i-1}{d-1}} \left(\lambda p\right)^{\frac{k_i-1}{d-1}} \\
&= \sum_{k_1=1}^{d} \sum_{k_2=k_1}^{dk_1} \sum_{k_3=k_2}^{dk_2} \cdots \sum_{k_i=k_{i-1}}^{dk_{i-1}} \binom{1}{\frac{k_1-1}{d-1}} \binom{k_1}{\frac{k_2-k_1}{d-1}} \cdots \binom{k_{i-1}}{\frac{k_i-k_{i-1}}{d-1}} \\
&\qquad\qquad \cdot \left(\lambda(1-p)\right)^{1+k_1+k_2+\cdots+k_{i-1}-\frac{k_i-1}{d-1}} \left(\lambda p\right)^{\frac{k_i-1}{d-1}} .
\end{aligned}
$$

□

Now that we have a closed form expression for the tail cdf of the queue length process, this result allows us to write the expected queue length $\mathbb{E}[Q]$ explicitly as well.

$$
\begin{aligned}
&\mathbb{E}[Q] \\
=&\sum_{i=1}^{\infty} \sum_{k_1=1}^{d} \sum_{k_2=k_1}^{dk_1} \sum_{k_3=k_2}^{dk_2} \cdots \sum_{k_i=k_{i-1}}^{dk_{i-1}} \binom{1}{\frac{k_1-1}{d-1}} \binom{k_1}{\frac{k_2-k_1}{d-1}} \cdots \binom{k_{i-1}}{\frac{k_i-k_{i-1}}{d-1}} \\
&\qquad\qquad \cdot \left(\lambda(1-p)\right)^{1+k_1+k_2+\cdots+k_{i-1}-\frac{k_i-1}{d-1}} \left(\lambda p\right)^{\frac{k_i-1}{d-1}} \\
\approx&\sum_{i=1}^{i^*} \sum_{k_1=1}^{d} \sum_{k_2=k_1}^{dk_1} \sum_{k_3=k_2}^{dk_2} \cdots \sum_{k_i=k_{i-1}}^{dk_{i-1}} \binom{1}{\frac{k_1-1}{d-1}} \binom{k_1}{\frac{k_2-k_1}{d-1}} \cdots \binom{k_{i-1}}{\frac{k_i-k_{i-1}}{d-1}} \\
&\qquad\qquad \cdot \left(\lambda(1-p)\right)^{1+k_1+k_2+\cdots+k_{i-1}-\frac{k_i-1}{d-1}} \left(\lambda p\right)^{\frac{k_i-1}{d-1}}
\end{aligned}
$$

where i^* is the smallest x such that $\mathbb{P}\left(Q \geq x\right) < \epsilon$.

In Table 4.2, we provide a table of mean queue lengths as a function of the flexibility parameter p and the choice parameter d when $\lambda = 0.99$. We observe that for $d = 2$, the mean queue is decreased by 30% by having 1% of the customers be flexible and a 75% reduction in mean queue length for 10 % of the customers being flexible. Thus, just a small amount of flexibility can go a long way. We also observe that these dramatic improvements are only strengthened when we increase the choice parameter d.

To study the impact of the flexibility and choice parameters on the fluctuations, we provide a table in Table 4.3 that describes the variance of the queue length as a function of the flexibility parameter p and the choice parameter d when $\lambda = 0.99$. We observe that for $d = 2$, the variance of the queue length is decreased by 65% by having 1% of the customers be flexible and a 97% reduction in variance queue length for 10 % of the customers being flexible. Thus, the reduction in variance is even better than the mean. Once again just a small amount of flexibility can significantly impact the performance of the system. We also observe for the variance that performance improvements increase when we increase the choice parameter d.

	p										
d	0	0.01	0.05	0.1	0.2	0.5	0.8	0.9	0.95	0.99	1
2	99	69	36	24	15	8	6	6	6	5	5
3	99	62	28	18	11	6	4	4	4	4	4
4	99	59	25	15	9	5	4	3	3	3	3
5	99	57	23	14	8	4	3	3	3	3	3
10	99	53	20	12	7	3	3	2	2	2	2
20	99	51	18	10	6	3	2	2	2	2	2
50	99	50	17	10	5	3	2	2	2	2	2
100	99	50	17	9	5	2	2	1	1	1	1

Table 4.2: $\mathbb{E}(Q)$ for various values of p and d

	p										
d	0	0.01	0.05	0.1	0.2	0.5	0.8	0.9	0.95	0.99	1
2	9890	3397	544	186	57	11	4	4	3	3	3
3	9890	2849	367	114	32	5	2	2	1	1	1
4	9890	2678	319	96	26	4	1	1	1	1	1
5	9890	2603	299	88	23	3	1	1	1	1	1
10	9890	2506	274	79	20	3	1	1	0	0	0
20	9890	2482	268	76	19	2	1	0	0	0	0
50	9890	2476	266	76	19	2	1	0	0	0	0
100	9890	2474	266	75	19	2	1	0	0	0	0

Table 4.3: var(Q) for various values of p and d

4.4.2 First and Second Moment Bounds

In this section, we prove upper and lower bounds for the mean, variance, and second moment of the queue length. We show numerically, that these bounds (especially the lower bounds) are quite accurate at approximating the queue length dynamics.

Proposition 8 (Moment Estimates)**.** *Let $\mathbb{E}[Q]$ denote the expected queue length, then*

$$\frac{\lambda\left(1+\frac{p\lambda^d}{1-\lambda^d(1-p)^d}\right)}{1-\lambda+\lambda p} < \mathbb{E}[Q] < \frac{\lambda\left(1+p\lambda^d\left(\frac{1-p}{1-\lambda^d}+\frac{p}{1-\lambda^{d^2}}\right)\right)}{1-\lambda+\lambda p}.$$

Let $\mathbb{E}[Q^2]$ denote the the second moment of queue length, then

$$\mathbb{E}[Q]^2 > \frac{\frac{2\lambda^{d+1}p}{(1-\lambda^d(1-p)^d)^2}+(1+\lambda(1-p))\frac{\lambda\left(1+\frac{p\lambda^d}{1-\lambda^d(1-p)^d}\right)}{1-\lambda+\lambda p}}{1-\lambda+\lambda p},$$

$$\mathbb{E}[Q^2] < \frac{2\lambda^{d+1}\left(\frac{1-p}{(1-\lambda^d)^2}+\frac{p}{(1-\lambda^{d^2})^2}\right)+(1+\lambda(1-p))\frac{\lambda\left(1+p\lambda^d\left(\frac{1-p}{1-\lambda^d}+\frac{p}{1-\lambda^{d^2}}\right)\right)}{1-\lambda+\lambda p}}{1-\lambda+\lambda p}.$$

Moreover, let W be the patient waiting time, then

$$\frac{1+\frac{p\lambda^d}{1-\lambda^d(1-p)^d}}{1-\lambda+\lambda p} < \mathbb{E}[W] < \frac{1+p\lambda^d\left(\frac{1-p}{1-\lambda^d}+\frac{p}{1-\lambda^{d^2}}\right)}{1-\lambda+\lambda p}.$$

Proof. Since we have the following bounds of the equilibrium s^I for $p \in (0,1)$ and $d \geq 2$,

$$\lambda^k(1-p)^{k-1} < s_k^I < \lambda^k, \quad k \geq 1. \tag{4.42}$$

Applying the above inequality to the recursion again, we get

$$s_k^I < \lambda(1-p)\lambda^{k-1} + \lambda p \lambda^{(k-1)d} = (1-p)\lambda^k + p\lambda^{(k-1)d+1}. \tag{4.43}$$

We can also bound the expected queue length the same way. Denote $x_i = s_i - s_{i+1}$ as the pdf of queue length Q, then

$$\begin{aligned}
\mathbb{E}[Q] &= \sum_{i=1}^{\infty} i x_i = \sum_{i=1}^{\infty} i(s_i^I - s_{i+1}^I) = \sum_{i=1}^{\infty} s_i^I \\
&= \sum_{i=0}^{\infty} \left[\lambda(1-p)s_i^I + \lambda p (s_i^I)^d\right] \\
&= \lambda(1-p)(\mathbb{E}[Q]+1) + \lambda p \left(1 + \sum_{i=1}^{\infty} (s_i^I)^d\right),
\end{aligned}$$

which implies that

$$\mathbb{E}[Q] = \frac{\lambda(1+pZ)}{1-\lambda+\lambda p} \tag{4.44}$$

where $Z = \sum_{i=1}^{\infty}(s_i^I)^d$. Thus, we can obtain an upper bound for Z,

$$\begin{aligned}
Z &= \sum_{i=1}^{\infty}(s_i^I)^d \\
\text{(Inequality (4.43))} \quad &< \sum_{i=1}^{\infty}((1-p)\lambda^i + p\lambda^{(i-1)d+1})^d \\
\text{(Jensen's Inequality)} \quad &\leq (1-p)\sum_{i=1}^{\infty}\lambda^{id} + p\sum_{i=1}^{\infty}\lambda^{((i-1)d+1)d} \\
&= \lambda^d\left(\frac{1-p}{1-\lambda^d} + \frac{p}{1-\lambda^{d^2}}\right).
\end{aligned}$$

Similarly we can obtain an lower bound for Z,

$$\begin{aligned} Z &= \sum_{i=1}^{\infty} (s_i^I)^d \\ \text{(Inequality (4.42))} &> \sum_{i=1}^{\infty} \left(\lambda^i (1-p)^{i-1}\right)^d \\ &= \frac{\lambda^d}{1-\lambda^d(1-p)^d}. \end{aligned}$$

Combined with equation (4.44), we obtain the upper and lower bound for $\mathbb{E}[Q]$ as follows,

$$\frac{\lambda\left(1+\frac{p\lambda^d}{1-\lambda^d(1-p)^d}\right)}{1-\lambda+\lambda p} < \mathbb{E}[Q] < \frac{\lambda\left(1+p\lambda^d\left(\frac{1-p}{1-\lambda^d}+\frac{p}{1-\lambda^{d^2}}\right)\right)}{1-\lambda+\lambda p}.$$

For the second moment, similarly we have that

$$\begin{aligned} \mathbb{E}[Q^2] &= \sum_{i=1}^{\infty} i^2 x_i = \sum_{i=1}^{\infty} i^2 (s_i^I - s_{i+1}^I) \\ &= \sum_{i=1}^{\infty} (i^2-(i-1)^2) s_i^I \\ &= 2\sum_{i=1}^{\infty} i s_i^I - \mathbb{E}[Q] \\ &= 2\sum_{i=0}^{\infty} (i+1)\left[\lambda(1-p)s_i^I + \lambda p (s_i^I)^d\right] - \mathbb{E}[Q] \\ &= 2\lambda(1-p)\left(\sum_{i=0}^{\infty} i s_i^I\right) + 2\mathbb{E}[Q] + 2\lambda p\left(\sum_{i=1}^{\infty} i(s_i^I)^d\right) - \mathbb{E}[Q] \end{aligned}$$

which implies that

$$\mathbb{E}[Q^2] = 2\sum_{i=1}^{\infty} i s_i^I - \mathbb{E}[Q], \tag{4.45}$$

and

$$\sum_{i=1}^{\infty} i s_i^I = \frac{\lambda p Z_2 + \mathbb{E}[Q]}{1-\lambda+\lambda p} \tag{4.46}$$

where $Z_2 = \sum_{i=1}^{\infty} i(s_i^I)^d$. We can obtain an upper bound for Z_2,

$$\begin{aligned} Z_2 &= \sum_{i=1}^{\infty} i(s_i^I)^d \\ \text{(Inequality (4.43))} &< \sum_{i=1}^{\infty} i((1-p)\lambda^i + p\lambda^{(i-1)d+1})^d \\ \text{(Jensen's Inequality)} &\leq (1-p)\sum_{i=1}^{\infty} i\lambda^{id} + p\sum_{i=1}^{\infty} i\lambda^{((i-1)d+1)d} \\ &= \lambda^d \left(\frac{1-p}{(1-\lambda^d)^2} + \frac{p}{(1-\lambda^{d^2})^2} \right). \end{aligned}$$

Similarly we can obtain an lower bound for Z_2,

$$\begin{aligned} Z_2 &= \sum_{i=1}^{\infty} i(s_i^I)^d \\ \text{(Inequality (4.42))} &> \sum_{i=1}^{\infty} i\left(\lambda^i(1-p)^{i-1}\right)^d \\ &= \frac{\lambda^d}{(1-\lambda^d(1-p)^d)^2}. \end{aligned}$$

Combined with Equations (4.45) and (4.46), we obtain the upper and lower bound for $\mathbb{E}[Q^2]$ as follows,

$$\mathbb{E}[Q^2] = 2 \cdot \frac{\lambda p Z_2 + \mathbb{E}[Q]}{1-\lambda+\lambda p} - \mathbb{E}[Q] = \frac{2\lambda p Z_2 + (1+\lambda(1-p))\mathbb{E}[Q]}{1-\lambda+\lambda p}$$

and

$$\mathbb{E}[Q^2] > \frac{\frac{2\lambda^{d+1}p}{(1-\lambda^d(1-p)^d)^2} + (1+\lambda(1-p))\frac{\lambda\left(1+\frac{p\lambda^d}{1-\lambda^d(1-p)^d}\right)}{1-\lambda+\lambda p}}{1-\lambda+\lambda p},$$

$$\mathbb{E}[Q^2] < \frac{2\lambda^{d+1}\left(\frac{1-p}{(1-\lambda^d)^2} + \frac{p}{(1-\lambda^{d^2})^2}\right) + (1+\lambda(1-p))\frac{\lambda\left(1+p\lambda^d\left(\frac{1-p}{1-\lambda^d}+\frac{p}{1-\lambda^{d^2}}\right)\right)}{1-\lambda+\lambda p}}{1-\lambda+\lambda p}.$$

If we use subscript U to denote upper bound and subscript L to denote lower bound, then we can obtain an upper bound for Var$[Q]$,

$$\text{Var}[Q] < \text{Var}_U[Q] = \mathbb{E}_U[Q^2] - \mathbb{E}_L[Q].$$

Similarly we can also obtain a lower bound for Var$[Q]$,

$$\text{Var}[Q] > \text{Var}_L[Q] = \mathbb{E}_L[Q^2] - \mathbb{E}_U[Q].$$

□

In Figure 4.6, we plot $\mathbb{E}[Q], \mathbb{E}[Q^2], \text{Var}[Q]$ as well as their upper and lower bounds obtained from Proposition 8. We note that our upper and lower bounds are quite accurate at approximating the moment behavior as a function of the flexibility parameter p. In Figure 4.5, we observe that the compare the wait times of dedicated patients, flexible patients, the average patients, and the system where all the flexible patients are not present. It is clear from Figure 4.5, that the average wait time decreases by adding flexible patients and the flexible patients benefit themselves from being flexible. Throughout our analysis, one might be tempted to approximate the queue length and waiting time with a model where the flexible patients disappeared i.e. a non-flexible queue where the arrival rate is $\lambda(1-p)$. However, we observe that the wait time is very much under-approximated if one pretends the flexible patients are not there. Thus, it is still important to capture the flexible patients and they cannot be simply ignored from the performance analysis.

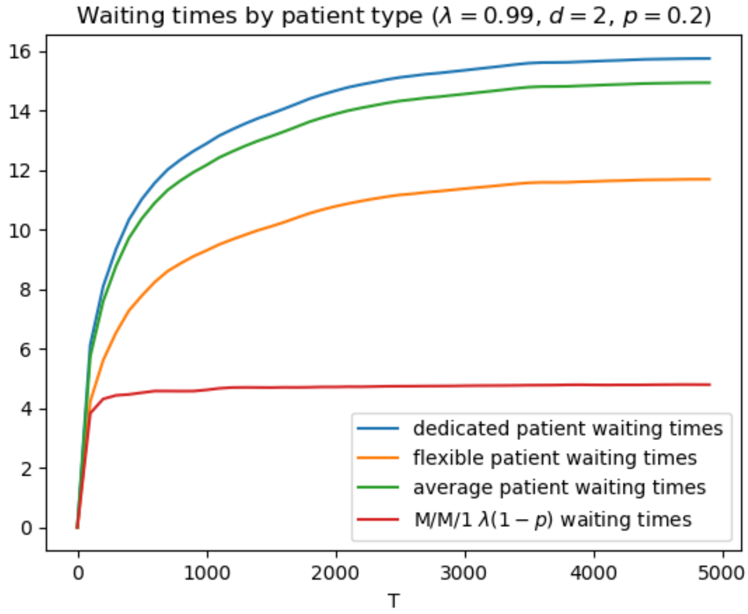


Figure 4.5: Waiting times by patient type

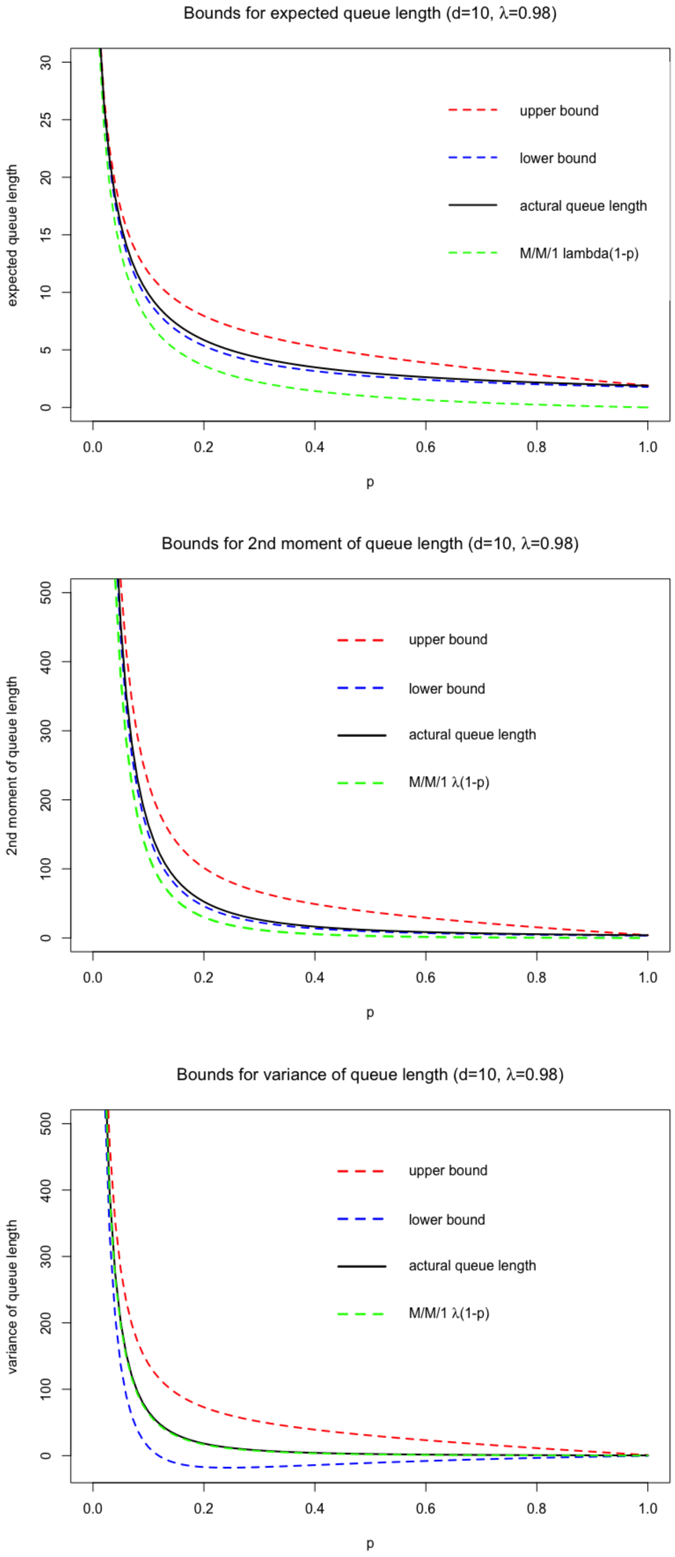


Figure 4.6: Upper and lower bounds for $\mathbb{E}(Q)$, $\mathbb{E}(Q^2)$, and $\mathrm{var}(Q)$

4.5 Discussion and Future Directions

In this chapter, we construct a stochastic queueing model that captures the performance trade-off between customers valuing flexibility (or time) vs. customers wanting dedicated services, through setting a fraction p of all customers to be flexible via joining the shortest of d queues. First, we prove the fluid model results in both transient and steady-state behaviors. We show that the scaled queue-length process converges to a unique fluid trajectory on any finite time interval, and that this fluid trajectory converges to a unique steady state s^I, for which a closed-form expression is obtained. We also show that the steady state distribution of the N-physicians system concentrates on s^I as N goes to infinity. Second, we prove the diffusion model results in both transient and steady-state behaviors. We show that the scaled diffusion process converges to a unique Ornstein-Uhlenbeck process, and that the interchanging of limits $\lim_{t\to\infty}\lim_{N\to\infty} = \lim_{N\to\infty}\lim_{t\to\infty}$ holds for the diffusion limit in equilibrium. Finally, we prove an upper and lower bound for the first and second moment of the expected queue length of the system, and show through numerical examples that having just a small fraction of flexible customers can benefit the system tremendously, both in lowering the mean queue length as well as its variance.

Despite our analysis, there are many future directions for research.

1. The first direction would be to generalize the arrival rate of dedicated patients to each physician to be non-uniform, i.e. taking into account the popularity of different physicians.
2. A second direction would be to generalize our results for non exponential arrival and service distributions, like in the work of Bramson et al. [15],

Aghajani et al. [3].

3. It would also be interesting to generalize the work to system of $M/M/c$ queues or system of $M/M/\infty$ queues, and derive new limit theorems in those regimes. One could also incorporate the impact of delayed information to model delays in communicating the queue length to customers. Recent work by Nirenberg et al. [66], Novitzky et al. [67] could help in this regard.
4. Finally, there is recent work that analyzes self-exciting point processes as arrival processes to queues, see for example Gao and Zhu [34], Daw and Pender [23], Koops et al. [49], Daw and Pender [22], Daw et al. [24], Chen and Wang [19]. It would be interesting to analyze similar JSQ models with Hawkes arrival processes.

We intend to pursue these extensions in future work.

BIBLIOGRAPHY

[1] https://www.zocdoc.com/about/blog/tech/how-zocdoc-improves-patient-wait-times/.

[2] https://money.cnn.com/interactive/economy/average-doctor-wait-times, 2018.

[3] Reza Aghajani, Xingjie Li, and Kavita Ramanan. The PDE method for the analysis of randomized load balancing networks. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 1(2):1–28, 2017. doi: 10.1145/3154497. URL https://doi.org/10.1145%2F3154497.

[4] Matteo Almanza, Flavio Chierichetti, Silvio Lattanzi, Alessandro Panconesi, and Giuseppe Re. Online facility location with multiple advice. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 4661–4673. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/file/250473494b245120a7eaf8b2e6b1f17c-Paper.pdf.

[5] Antonios Antoniadis, Themis Gouleakis, Pieter Kleer, and Pavel Kolev. Secretary and online matching problems with machine learned advice, 2020. URL https://arxiv.org/abs/2006.01026.

[6] Nilay Tanık Argon and Serhan Ziya. Priority assignment under imperfect information on customer type identities. *Manufacturing & Service Operations Management*, 11(4):674–693, 2009. doi: 10.1287/msom.1080.0246. URL https://doi.org/10.1287%2Fmsom.1080.0246.

[7] Yossi Azar, Stefano Leonardi, and Noam Touitou. Flow time scheduling with uncertain processing time. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*. ACM, 2021. doi: 10.1145/3406325.3451023. URL https://doi.org/10.1145%2F3406325.3451023.

[8] Egon Balas. On the facial structure of scheduling polyhedra. In *Mathematical Programming Essays in Honor of George B. Dantzig Part I*, pages 179–218. Springer Berlin Heidelberg, 1985. doi: 10.1007/bfb0121051. URL https://doi.org/10.1007%2Fbfb0121051.

[9] Eric Balkanski, Tingting Ou, Clifford Stein, and Hao-Ting Wei. Scheduling with speed predictions, 2022. URL https://arxiv.org/abs/2205.01247.

[10] Santiago Balseiro, Christian Kroer, and Rachitesh Kumar. Single-leg revenue management with advice, 2022. URL https://arxiv.org/abs/2202.10939.

[11] Sayan Banerjee and Debankur Mukherjee. Join-the-shortest queue diffusion limit in Halfin–Whitt regime: Tail asymptotics and scaling of extrema. *The Annals of Applied Probability*, 29(2), 2019. doi: 10.1214/18-aap1436. URL https://doi.org/10.1214%2F18-aap1436.

[12] Siddhartha Banerjee, Vasilis Gkatzelis, Artur Gorokh, and Billy Jin. Online nash social welfare maximization with predictions. In *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1–19. Society for Industrial and Applied Mathematics, 2022. doi: 10.1137/1.9781611977073.1. URL https://doi.org/10.1137%2F1.9781611977073.1.

[13] Mikhail Batsyn, Boris Goldengorin, Panos M. Pardalos, and Pavel Sukhov. Online heuristic for the preemptive single machine scheduling problem of minimizing the total weighted completion time. *Optimization Methods and Software*, 29(5):955–963, 2013. doi: 10.1080/10556788.2013.854360. URL https://doi.org/10.1080%2F10556788.2013.854360.

[14] Maury Bramson. Stability of join the shortest queue networks. *The Annals of Applied Probability*, 21(4), 2011. doi: 10.1214/10-aap726. URL https://doi.org/10.1214%2F10-aap726.

[15] Maury Bramson, Yi Lu, and Balaji Prabhakar. Decay of tails at equilibrium for FIFO join the shortest queue networks. *The Annals of Applied Probability*, 23(5), 2013. doi: 10.1214/12-aap888. URL https://doi.org/10.1214%2F12-aap888.

[16] Anton Braverman. Steady-state analysis of the join-the-shortest-queue model in the Halfin–Whitt regime. *Mathematics of Operations Research*, 45 (3):1069–1103, 2020. doi: 10.1287/moor.2019.1023. URL https://doi.org/10.1287%2Fmoor.2019.1023.

[17] Peter Brucker. *Scheduling Algorithms*. Springer Berlin Heidelberg, 2007. doi: 10.1007/978-3-540-69516-5. URL https://doi.org/10.1007%2F978-3-540-69516-5.

[18] Justin Chen, Sandeep Silwal, Ali Vakilian, and Fred Zhang. Faster fundamental graph algorithms via learned predictions. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 3583–

3602. PMLR, 2022. URL https://proceedings.mlr.press/v162/chen22v.html.

[19] Xinyun Chen and Xiuwen Wang. Perfect sampling of multivariate Hawkes processes. In *2020 Winter Simulation Conference (WSC)*. IEEE, 2020. doi: 10.1109/wsc48552.2020.9384038. URL https://doi.org/10.1109%2Fwsc48552.2020.9384038.

[20] Richard Walter Conway, William L. Maxwell, and Louis W. Miller. *Theory of Scheduling*. Addison-Wesley Publishing Company, 1967.

[21] J. G. Dai, John J. Hasenbein, and Bara Kim. Stability of join-the-shortest-queue networks. *Queueing Systems*, 57(4):129–145, 2007. doi: 10.1007/s11134-007-9046-5. URL https://doi.org/10.1007%2Fs11134-007-9046-5.

[22] Andrew Daw and Jamol Pender. Exact simulation of the queue-hawkes process. In *Proceedings of the 2018 Winter Simulation Conference*, pages 4234–4235. IEEE Press, 2018.

[23] Andrew Daw and Jamol Pender. Queues driven by Hawkes processes. *Stochastic Systems*, 8(3):192–229, 2018. doi: 10.1287/stsy.2018.0014. URL https://doi.org/10.1287%2Fstsy.2018.0014.

[24] Andrew Daw, Antonio Castellanos, Galit B. Yom-Tov, Jamol Pender, and Leor Gruendlinger. The co-production of service: Modeling service times in contact centers using Hawkes processes. *CoRR*, 2020. doi: 10.48550/ARXIV.2004.07861. URL https://arxiv.org/abs/2004.07861.

[25] A. B. Dieker and T. Suk. Randomized longest-queue-first scheduling for large-scale buffered systems. *Advances in Applied Probability*, 47(4):1015–

1038, 2015. doi: 10.1239/aap/1449859798. URL https://doi.org/10.1239%2Faap%2F1449859798.

[26] Michael Dinitz, Sungjin Im, Thomas Lavastida, Benjamin Moseley, and Sergei Vassilvitskii. Faster matchings via learned duals, 2021. URL https://arxiv.org/abs/2107.09770.

[27] Erik A. Van Doorn. Conditions for exponential ergodicity and bounds for the decay parameter of a birth-death process. *Advances in Applied Probability*, 17(3):514–530, 1985. doi: 10.2307/1427118. URL https://doi.org/10.2307%2F1427118.

[28] Paul Dütting, Silvio Lattanzi, Renato Paes Leme, and Sergei Vassilvitskii. Secretaries with advice. In *Proceedings of the 22nd ACM Conference on Economics and Computation*. ACM, 2021. doi: 10.1145/3465456.3467623. URL https://doi.org/10.1145%2F3465456.3467623.

[29] W. L. Eastman, S. Even, and I. M. Isaacs. Bounds for the optimal scheduling of n jobs on m processors. *Management Science*, 11(2):268–279, 1964. doi: 10.1287/mnsc.11.2.268. URL https://doi.org/10.1287/mnsc.11.2.268.

[30] Patrick Eschenfeldt and David Gamarnik. Join the shortest queue with many servers. The heavy-traffic asymptotics. *Mathematics of Operations Research*, 43(3):867–886, 2018. doi: 10.1287/moor.2017.0887. URL https://doi.org/10.1287%2Fmoor.2017.0887.

[31] Stewart N. Ethier and Thomas G. Kurtz. *Markov processes: characterization and convergence*, volume 282. John Wiley & Sons, 2009.

[32] R. D. Foley and D. R. McDonald. Join the shortest queue: stability and exact asymptotics. *The Annals of Applied Probability*, 11(3), 2001. doi: 10.1214/aoap/1015345342. URL https://doi.org/10.1214%2Faoap%2F1015345342.

[33] S. Foss and A. L. Stolyar. Large-scale join-idle-queue system with general service times. *Journal of Applied Probability*, 54(4):995–1007, 2017. doi: 10.1017/jpr.2017.49. URL https://doi.org/10.1017%2Fjpr.2017.49.

[34] Xuefeng Gao and Lingjiong Zhu. Functional central limit theorems for stationary Hawkes processes and application to infinite-server queues. *Queueing Systems*, 90(1-2):161–206, 2018. doi: 10.1007/s11134-018-9570-5. URL https://doi.org/10.1007%2Fs11134-018-9570-5.

[35] Michel X. Goemans and David P. Williamson. Two-dimensional Gantt charts and a scheduling algorithm of Lawler. *SIAM Journal on Discrete Mathematics*, 13(3):281–294, 2000. doi: 10.1137/S0895480197330254. URL https://doi.org/10.1137/S0895480197330254.

[36] Carl Graham. Chaoticity on path space for a queueing network with selection of the shortest queue among several. *Journal of Applied Probability*, 37(1):198–211, 2000. ISSN 00219002. URL http://www.jstor.org/stable/3215668.

[37] Carl Graham. Kinetic limits for large communication networks. In *Modeling in Applied Sciences*, pages 317–370. Birkhäuser Boston, 2000. doi: 10.1007/978-1-4612-0513-5_9. URL https://doi.org/10.1007%2F978-1-4612-0513-5_9.

[38] Carl Graham. Chaoticity results for "join the shortest queue". *Contemporary*

Mathematics, 275:53–68, 2001. doi: 10.1090/conm/275/04490. URL https://doi.org/10.1090%2Fconm%2F275%2F04490.

[39] Carl Graham. Functional central limit theorems for a large network in which customers join the shortest of several queues. *Probability Theory and Related Fields*, 131(1):97–120, 2004. doi: 10.1007/s00440-004-0372-9. URL https://doi.org/10.1007%2Fs00440-004-0372-9.

[40] Ronald L. Graham, Eugene L. Lawler, Jan Karel Lenstra, and Alexander Hendrik George Rinnooy Kan. Optimization and approximation in deterministic sequencing and scheduling: A survey. In *Discrete Optimization II*, volume 5 of *Annals of Discrete Mathematics*, pages 287–326. Elsevier, 1979. doi: https://doi.org/10.1016/S0167-5060(08)70356-X. URL https://www.sciencedirect.com/science/article/pii/S016750600870356X.

[41] Leslie A. Hall and Fabian Chudak. Private communication, 1997.

[42] Leslie A. Hall, Andreas S. Schulz, David B. Shmoys, and Joel Wein. Scheduling to minimize average completion time: Off-line and on-line approximation algorithms. *Mathematics of Operations Research*, 22(3):513–544, 1997. doi: 10.1287/moor.22.3.513. URL https://doi.org/10.1287/moor.22.3.513.

[43] Yu-Tong He and Douglas G. Down. Limited choice and locality considerations for load balancing. *Performance Evaluation*, 65(9):670–687, 2008. doi: 10.1016/j.peva.2008.03.001. URL https://doi.org/10.1016%2Fj.peva.2008.03.001.

[44] Sungjin Im, Ravi Kumar, Mahshid Montazer Qaem, and Manish

Purohit. Online knapsack with frequency predictions. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 2733–2743. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/file/161c5c5ad51fcc884157890511b3c8b0-Paper.pdf.

[45] Sungjin Im, Ravi Kumar, Mahshid Montazer Qaem, and Manish Purohit. Non-clairvoyant scheduling with predictions. In *Proceedings of the 33rd ACM Symposium on Parallelism in Algorithms and Architectures*. ACM, 2021. doi: 10.1145/3409964.3461790. URL https://doi.org/10.1145%2F3409964.3461790.

[46] Samuel Karlin and James L. McGregor. The classification of birth and death processes. *Transactions of the American Mathematical Society*, 86(2):366–400, 1957. doi: 10.1090/s0002-9947-1957-0094854-8. URL https://doi.org/10.1090%2Fs0002-9947-1957-0094854-8.

[47] Samuel Karlin and James L. McGregor. The differential equations of birth-and-death processes, and the Stieltjes moment problem. *Transactions of the American Mathematical Society*, 85(2):489–546, 1957. doi: 10.1090/s0002-9947-1957-0091566-1. URL https://doi.org/10.1090%2Fs0002-9947-1957-0091566-1.

[48] Neal Klitsch. How I read imaging studies, 2016. URL http://www.neighborhoodradiologist.com/how-i-read-imaging-studies/.

[49] D. T. Koops, M. Saxena, O. J. Boxma, and M. Mandjes. Infinite-server queues with Hawkes input. *Journal of Applied Probability*, 55(3):920–943,

2018. doi: 10.1017/jpr.2018.58. URL https://doi.org/10.1017%2Fjpr.2018.58.

[50] Silvio Lattanzi, Thomas Lavastida, Benjamin Moseley, and Sergei Vassilvitskii. Online scheduling via learned weights. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1859–1877. Society for Industrial and Applied Mathematics, 2020. doi: 10.1137/1.9781611975994.114. URL https://doi.org/10.1137%2F1.9781611975994.114.

[51] Thomas Lavastida, Benjamin Moseley, R. Ravi, and Chenyang Xu. Using predicted weights for ad delivery. In *SIAM Conference on Applied and Computational Discrete Algorithms (ACDA21)*, pages 21–31. Society for Industrial and Applied Mathematics, 2021. doi: 10.1137/1.9781611976830.3. URL https://doi.org/10.1137%2F1.9781611976830.3.

[52] Eugene L. Lawler, Jan Karel Lenstra, Alexander Hendrik George Rinnooy Kan, and David B. Shmoys. Chapter 9 Sequencing and scheduling: Algorithms and complexity. In *Logistics of Production and Inventory*, volume 4 of *Handbooks in Operations Research and Management Science*, pages 445–522. Elsevier, 1993. doi: 10.1016/s0927-0507(05)80189-6. URL https://doi.org/10.1016%2Fs0927-0507%2805%2980189-6.

[53] Jan Karel Lenstra and David B. Shmoys. Elements of scheduling. *CoRR*, 2020. URL https://arxiv.org/abs/2001.06005.

[54] Hwa-Chun Lin and C.S. Raghavendra. An approximate analysis of the join the shortest queue (JSQ) policy. *IEEE Transactions on Parallel and Distributed Systems*, 7(3):301–307, 1996. doi: 10.1109/71.491583. URL https://doi.org/10.1109%2F71.491583.

[55] Alexander Lindermayr and Nicole Megow. Permutation predictions for non-clairvoyant scheduling, 2022. URL https://arxiv.org/abs/2202.10199.

[56] Yi Lu, Qiaomin Xie, Gabriel Kliot, Alan Geller, James R. Larus, and Albert Greenberg. Join-idle-queue: A novel load balancing algorithm for dynamically scalable web services. *Performance Evaluation*, 68(11):1056–1071, 2011. doi: 10.1016/j.peva.2011.07.015. URL https://doi.org/10.1016%2Fj.peva.2011.07.015.

[57] Thodoris Lykouris and Sergei Vassilvitskii. Competitive caching with machine learned advice. *Journal of the ACM*, 68(4):1–25, 2021. doi: 10.1145/3447579. URL https://doi.org/10.1145%2F3447579.

[58] Laura A. McLay and Maria E. Mayorga. A model for optimally dispatching ambulances to emergency calls with classification errors in patient priorities. *IIE Transactions*, 45(1):1–24, 2013. doi: 10.1080/0740817x.2012.665200. URL https://doi.org/10.1080%2F0740817x.2012.665200.

[59] Michael Mitzenmacher. Studying balanced allocations with differential equations. *Combinatorics, Probability and Computing*, 8(5):473–482, 1999. doi: 10.1017/s0963548399003946. URL https://doi.org/10.1017%2Fs0963548399003946.

[60] Michael Mitzenmacher. The power of two choices in randomized load balancing. *IEEE Transactions on Parallel and Distributed Systems*, 12(10):1094–1104, 2001. doi: 10.1109/71.963420. URL https://doi.org/10.1109%2F71.963420.

[61] Michael Mitzenmacher. Scheduling with predictions and the price of

misprediction. 2020. doi: 10.4230/LIPICS.ITCS.2020.14. URL https://drops.dagstuhl.de/opus/volltexte/2020/11699/.

[62] Michael Mitzenmacher and Sergei Vassilvitskii. Algorithms with predictions. *CoRR*, arXiv: https://arxiv.org/abs/2006.09123, 2020. doi: 10.48550/ARXIV.2006.09123. URL https://arxiv.org/abs/2006.09123.

[63] Maxim Mokin, Sameer A Ansari, Ryan A McTaggart, Ketan R Bulsara, Mayank Goyal, Michael Chen, and Justin F Fraser. Indications for thrombectomy in acute ischemic stroke from emergent large vessel occlusion (ELVO): report of the SNIS standards and guidelines committee. *Journal of NeuroInterventional Surgery*, 11(3):215–220, 2019. doi: 10.1136/neurintsurg-2018-014640. URL https://doi.org/10.1136%2Fneurintsurg-2018-014640.

[64] Debankur Mukherjee, Sem C. Borst, Johan S. H. van Leeuwaarden, and Philip A. Whiting. Universality of power-of-d load balancing schemes. *ACM SIGMETRICS Performance Evaluation Review*, 44(2):36–38, 2016. doi: 10.1145/3003977.3003990. URL https://doi.org/10.1145%2F3003977.3003990.

[65] Debankur Mukherjee, Sem C. Borst, Johan S. H. van Leeuwaarden, and Philip A. Whiting. Universality of power-of-d load balancing in many-server systems. *Stochastic Systems*, 8(4):265–292, 2018. doi: 10.1287/stsy.2018.0016. URL https://doi.org/10.1287%2Fstsy.2018.0016.

[66] Samantha Nirenberg, Andrew Daw, and Jamol Pender. THE IMPACT OF QUEUE LENGTH ROUNDING AND DELAYED APP INFORMATION ON DISNEY WORLD QUEUES. In *2018 Winter Simulation Confer-*

ence (WSC). IEEE, 2018. doi: 10.1109/wsc.2018.8632436. URL https://doi.org/10.1109%2Fwsc.2018.8632436.

[67] Sophia Novitzky, Jamol Pender, Richard H. Rand, and Elizabeth Wesson. Nonlinear dynamics in queueing theory: Determining the size of oscillations in queues with delay. *SIAM Journal on Applied Dynamical Systems*, 18 (1):279–311, 2019. doi: 10.1137/18m1170637. URL https://doi.org/10.1137%2F18m1170637.

[68] Christos H. Papadimitriou and Kenneth Steiglitz. *Combinatorial optimization: Algorithms and complexity*. Prentice Hall, 1982. ISBN 0131524623 9780131524620 0486402584 9780486402581.

[69] Manish Purohit, Zoya Svitkina, and Ravi Kumar. Improving online algorithms via ml predictions. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31, 2018. URL https://proceedings.neurips.cc/paper/2018/file/73a427badebe0e32caa2e1fc7530b7f3-Paper.pdf.

[70] Maurice Queyranne. Structure of a simple scheduling polyhedron. *Mathematical Programming*, 58(1):263–285, 1993. doi: 10.1007/BF01581271. URL https://doi.org/10.1007/BF01581271.

[71] Maurice Queyranne and Andreas S. Schulz. Polyhedral approaches to machine scheduling. Technical report, 1994.

[72] Maurice Queyranne and Yaoguang Wang. Single-machine scheduling polyhedra with precedence constraints. *Mathematics of Operations Research*,

16(1):1–20, 1991. ISSN 0364765X, 15265471. URL http://www.jstor.org/stable/3689846.

[73] S. Rabinovich, G. Berkolaiko, and S. Havlin. Solving nonlinear recursions. *Journal of Mathematical Physics*, 37(11):5828–5836, 1996. doi: 10.1063/1.531702. URL https://doi.org/10.1063%2F1.531702.

[74] Linus Schrage. A proof of the optimality of the shortest remaining processing time discipline. *Operations Research*, 16(3):687–690, 1968. doi: 10.1287/opre.16.3.687. URL https://doi.org/10.1287%2Fopre.16.3.687.

[75] Andreas S. Schulz. *Polytopes and Scheduling*. PhD thesis, TU Berlin, 1996.

[76] George Shih. Private communication, 2022.

[77] Simrita Singh, Itai Gurvich, and Jan Albert Van Mieghem. Feature-based design of priority queues: Digital triage in healthcare. *SSRN Electronic Journal*, 2020. doi: 10.2139/ssrn.3731865. URL https://doi.org/10.2139%2Fssrn.3731865.

[78] Wayne E. Smith. Various optimizers for single-stage production. *Naval Research Logistics Quarterly*, 3(1-2):59–66, 1956. doi: 10.1002/nav.3800030106. URL https://doi.org/10.1002%2Fnav.3800030106.

[79] Shuang Tao and Jamol Pender. A STOCHASTIC ANALYSIS OF BIKE-SHARING SYSTEMS. *Probability in the Engineering and Informational Sciences*, 35(4):781–838, 2020. doi: 10.1017/s0269964820000297. URL https://doi.org/10.1017%2Fs0269964820000297.

[80] Yee Lam Elim Thompson, Gary Levine, Weijie Chen, Berkman Sahiner, Qin Li, Nicholas Petrick, and Frank W. Samuelson. Wait-time-saving analysis and clinical effectiveness of computer-aided triage and notification

(CADt) devices based on queueing theory. In Claudia R. Mello-Thoms and Sian Taylor-Phillips, editors, *Medical Imaging 2022: Image Perception, Observer Performance, and Technology Assessment*. SPIE, 2022. doi: 10.1117/12.2603184. URL https://doi.org/10.1117%2F12.2603184.

[81] John N. Tsitsiklis and Kuang Xu. On the power of (even a little) resource pooling. *Stochastic Systems*, 2(1):1–66, 2012. doi: 10.1287/11-ssy033. URL https://doi.org/10.1287%2F11-ssy033.

[82] Stephen R. E. Turner. The effect of increasing routing choice on resource pooling. *Probability in the Engineering and Informational Sciences*, 12(1):109–124, 1998. doi: 10.1017/s0269964800005088. URL https://doi.org/10.1017%2Fs0269964800005088.

[83] S. P. van der Zee and H. Theil. Priority assignment in waiting-line problems under conditions of misclassification. *Operations Research*, 9(6):875–885, 1961. doi: 10.1287/opre.9.6.875. URL https://doi.org/10.1287%2Fopre.9.6.875.

[84] Nikita Dmitrievna Vvedenskaya, Roland L'vovich Dobrushin, and Fridrikh Izrailevich Karpelevich. Queueing system with selection of the shortest of two queues: An asymptotic approach. *Probl. Peredachi Inf.*, 32 (1):20–34, 1996.

[85] Ward Whitt. Blocking when service is required from several facilities simultaneously. *AT&T Technical Journal*, 64(8):1807–1856, 1985. doi: 10.1002/j.1538-7305.1985.tb00038.x. URL https://doi.org/10.1002%2Fj.1538-7305.1985.tb00038.x.

[86] Laurence A. Wolsey. Mixed integer programming formulations for produc-

tion planning and scheduling problems. In *12th International Symposium on Mathematical Programming*, 1985.

[87] Bo Xiong and Christine Chung. Completion time scheduling and the WSRPT algorithm. In *Lecture Notes in Computer Science*, pages 416–426. Springer Berlin Heidelberg, 2012. doi: 10.1007/978-3-642-32147-4_37. URL https://doi.org/10.1007%2F978-3-642-32147-4_37.

[88] Chenyang Xu and Pinyan Lu. Mechanism design with predictions, 2022. URL https://arxiv.org/abs/2205.11313.