TWO PROBLEMS IN RESOURCE MANAGEMENT

TWO PROBLEMS IN RESOURCE MANAGEMENT: SCHEDULING WITH PREDICTION ERRORS AND FAIR DATA-DRIVEN ALLOCATION WITH LIMITED DATA

BY

MARYAM AKBARI-MOGHADDAM, B.Sc.

A THESIS

SUBMITTED TO THE DEPARTMENT OF COMPUTING AND SOFTWARE

AND THE SCHOOL OF GRADUATE STUDIES

OF MCMASTER UNIVERSITY

IN PARTIAL FULFILMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF SCIENCE

Master of Science (2021) McMaster University

(Computing and Software) Hamilton, Ontario, Canada

TITLE:	Two Problems in Resource Management: Scheduling with Prediction Errors and Fair Data-Driven Allocation with Limited Data
AUTHOR:	Maryam Akbari-Moghaddam B.Sc. (Information Technology), McMaster University, Hamilton, Canada
SUPERVISOR:	Dr. Douglas G. Down and Dr. Na Li
NUMBER OF PAGES:	x, 89

Abstract

Resource management is challenging when one needs to allocate scarce or limited resources to different entities with heterogeneous demands. In many practical situations, predictions of relevant quantities are only possible or available. While accurate estimates can certainly allow for better decisions to be made, a key challenge is to extract the maximum benefit when highly accurate estimates are not available (or possible). Even in the presence of reasonable estimates, other factors such as the need for a timely or real-time resource allocation can add to the complexity of the resource management process. This thesis studies two problems in resource management: scheduling with prediction errors and fair data-driven allocation with limited data. These problems both consider scenarios where only estimates of the demand are known and real-time resource allocation is required. In the second problem, the supply for resources is also not known within the decision-making period and is estimated.

The first problem considers a single-server queue that needs to schedule jobs without knowing the exact processing times. The processor, the limited resource, needs to be utilized by each job until it completes. The goal is to minimize the mean sojourn time of the system, which is the mean time between jobs' arrival and their completion. In practical settings, knowing the exact processing times of the jobs is not possible; however, estimates of the job sizes can be calculated. We introduce a heuristic that only uses estimated processing times for scheduling decisions and thus requires minimal calculation overhead. SEH does

not rely on any information that might not be available in real-world situations, such as the job processing time and estimation error distributions. We demonstrate that SEH shows desirable performance in minimizing the mean sojourn time of the system when jobs exhibit estimation error distribution variance that is consistent with that seen in practical settings.

In the second problem, we tackle the issue of resource allocation during epidemics when the resources are often scarce, in high demand, and need to be allocated in a timely manner. We discuss a model that is suitable for short-term real-time supply and demand forecasting during emerging epidemics without having to rely on demographic information. A data-driven resource allocation model is then proposed that minimizes a notion of fairness among the demand entities. We study the application of our model in a COVID-19 convalescent plasma (CCP) case study and provide numerical results that support the performance of our model in allocating the scarce CCP in a fair manner. Our results are close to the scenario where the exact supply and demand are known and more efficient than what was performed in practice.

To my parents,

whose endless support always gave me courage,

and lighted up my way.

Acknowledgements

I would like to acknowledge my co-supervisor, Dr. Douglas G. Down, for being all I could ever wish to find in a supervisor. Working as a graduate student under his supervision was the best experience of my whole academic life. His sincere support, patience and guidance were a significant part of my accomplishments during my program. I have felt fully supported from the beginning of this research and had peace of mind knowing that he was always willing to help and open to hearing my ideas. Working with him was a great pleasure, and I would like to express my deepest gratitude.

I would also like to show my profound gratitude to my co-supervisor, Dr. Na Li, whose suggestions and invaluable insights have been a great help in my research. She was always approachable and willing to listen to my numerous questions. She greatly encouraged and guided me both in my program and my future direction. I am most thankful for her caring, kind, and patient personality.

I am immensely grateful to my parents, Faraz and Mahnaz, to whom I owe all that I am, or ever hope to be. I would also like to acknowledge my brother, Parsa, for his genuine encouragement and Sahand Akbari for his endless support and help.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Resources are anything that is required for executing a task, and they need to be managed to maximize efficiency in an organization. The process of pre-planning, scheduling, and allocating the resources to achieve maximum efficiency is called resource management. Various challenges are associated with the proper allocation of resources. To name a few, first, the exact information about the resource, such as its needed amount or the time that it should be allocated to a specific task, is often not available until the task is completed. However, partial information may be present that can assist with estimating this information and help the allocation process. Second, resources must be assigned to different tasks or events in a timely manner to be effective. Third, resources might be scarce or very limited compared to the amount that they are requested. Furthermore, different entities might have heterogeneous demands for resources that must be considered in the planning phase. Thus, making reasonable supply and/or estimates for resources can have a significant impact on the resource allocation process and becomes challenging when the available data is very limited. In this work, we study two problems in resource management: scheduling with prediction errors and fair data-driven allocation with limited data. The two problems are

proposed in two papers, one published and one submitted, and are discussed in Chapter 2 and Chapter 3, respectively, which we now outline.

Chapter 2 includes the first problem, where we study scheduling a single-server system when exact information about the jobs' processing times is not available. Scheduling policies and their performance evaluation in a preemptive single-server queue have been a subject of interest for some time. Size-based policies are known to perform better than size-oblivious policies (that do not use any information about the exact job sizes) with respect to sojourn time, the time between a job's arrival to the system and its completion. For a single-server system, Shortest Remaining Processing Time (SRPT) is an optimal size-based policy. However, size-based policies such as SRPT are rarely deployed in practical settings. A key disadvantage is that when the exact processing times are not known to the system before scheduling, which is often the case in practical settings, their performance may significantly degrade. Most existing size-based policies in the literature rely on knowing the job processing time and estimation error distributions before scheduling. The assumption of knowing these distributions before scheduling may be problematic in real environments and formulating a policy under these assumptions introduces computational overhead that may be prohibitive. Our work assumes that the processing time is not available to the processor until the job is fully processed, but that processing time estimations are available.

We propose a simple heuristic, Size Estimate Hedging (SEH), that combines the merits of two size-based policies, Shortest Estimated Remaining Processing Time (SERPT) and Shortest Estimated Processing Time (SEPT). SERPT is a version of SRPT that employs estimated processing times and schedules jobs based on their estimated remaining times, and SEPT is a version of the Shortest Processing Time (SPT) policy that skips updating the estimated remaining processing times and prioritizes jobs based only on their estimated

processing times.

SEH requires minimal calculation overhead and no information about the job processing time and estimation error distributions. In other words, SEH only uses estimated processing times for scheduling decisions. A job's priority under SEH is increased dynamically according to an SRPT rule until it is determined that it is underestimated, at which time the priority is frozen. We compare the performance of our heuristic with existing policies in the literature for scheduling jobs in the presence of inexact size estimates and provide numerical results obtained by running a wide range of simulations for both synthetic and real workloads. We consider two performance metrics, mean sojourn time (MST) and mean slowdown, for reporting our results. We show that SEH has desirable performance in minimizing both the MST and mean slowdown of the system when there is sufficiently low variance in the estimation error distribution, a situation that is consistent with what is seen in practice.

In Chapter 3, we study our second problem, fair data-driven allocation with limited data. We focus on epidemics, which have impacted the world many times, and will occur again in the future. Timely responses during epidemics have a great role in minimizing the difficulties introduced as a result of the fast and widespread occurrence of an infectious disease in a society. However, during emerging epidemics, which are usually unexpected and can spread rapidly, there are often limited resources that have the ability to mitigate the effects of the disease. The allocation of these resources in a fair manner becomes more challenging as the total number of entities requesting them outnumbers the available resources. Furthermore, several other key reasons are of concern for decision-makers, including not having sufficient prior information about the disease, and dealing with a periodically changing and location-specific disease behaviour that can arise naturally or

as a result of government policies. These issues motivate the investigation of supply and demand forecasting models for scenarios where there are small amounts of available data, and the data can exhibit fundamental changes in behaviour. It is of interest to incorporate these models into algorithms that yield fair allocations.

We discuss real-time short-term forecasting of supply and demand of scarce resources in epidemics with very sparse and limited data, no historical data and without relying on epidemiological models or demographic information. We discuss and choose a forecasting model that does not require indeterminate parameters (such as location and time-specific parameters) and thus does not require periodically updating the parameters. We then address the challenges that may arise in an online setting due to extrapolation and sparse data by suggesting potential solutions. Next, we propose a data-driven MIP model for real-time multi-location allocation of scarce resources regularly and fairly to the entities requesting them, where the demand is heterogeneous and arises from geographically dispersed locations. This approach maximizes a notion of fairness among the resource-demanding entities. Numerical results obtained from a COVID-19 Convalescent Plasma (CCP) case study suggest that our approach can help minimize the unmet CCP demand ratios and lead to balanced and fair CCP allocation decisions. We show that these fair allocations are both close to the scenario where supply and demand are known (rather than forecast) and are preferable to what was used in practice.

Finally, in Chapter 4, we conclude our achievements and discuss possible directions for future work.

Chapter 2

SEH: Size Estimate Hedging for Single-Server Queues

This chapter is adapted from Maryam Akbari-Moghaddam and Douglas G. Down, "SEH: Size estimate hedging for single-server queues", 18th International Conference on Quantitative Evaluation of Systems (QEST 2021) (1). Complementary explanations on deriving the formulas are included in Section 2.3.5.

Abstract

For a single server system, Shortest Remaining Processing Time (SRPT) is an optimal size-based policy. In this chapter, we discuss scheduling a single-server system when exact information about the jobs' processing times is not available. When the SRPT policy uses estimated processing times, the underestimation of large jobs can significantly degrade performance. We propose a simple heuristic, Size Estimate Hedging (SEH), that only uses estimated processing times for scheduling decisions. A job's priority is increased dynamically according to an SRPT rule until it is determined that it is underestimated,

at which time the priority is frozen. Numerical results suggest that SEH has desirable performance for estimation error variance that is consistent with what is seen in practice.

Keywords: Estimated Job Sizes, M/G/1, Gittins' Index Policy, Size Estimate Hedging

2.1 Introduction

Over the past decades, there has been significant study on the scheduling of jobs in single-server queues. When preemption is allowed and processing times are known to the scheduler, the Shortest Remaining Processing Time (SRPT) policy is optimal in the sense that, regardless of the processing time distribution, it minimizes the number of jobs in the system at each point in time and hence, minimizes the mean sojourn time (MST) (2), (3). However, scheduling policies such as SRPT are rarely deployed in practical settings. A key disadvantage is that the assumption of knowing the exact job processing times prior to scheduling is not always practical to make. However, it is often possible to estimate the job processing times and use this approximate information for scheduling. The Shortest Estimated Remaining Processing Time (SERPT) policy is a version of SRPT that employs the job processing time estimates as if they were error-free and thus, schedules jobs based on their estimated remaining times. Motivated by the fact that estimates can often be obtained through machine learning techniques, Mitzenmacher (4) studies the potential benefits of using such estimates for simple scheduling policies. For this purpose, a price for misprediction, the ratio between a job's expected sojourn time using its estimated processing time and the job's expected sojourn time when the job processing time is known is introduced, and a bound on this price is given. The results in (4) suggest that naïve policies work well, and even a weak predictor

can yield significant improvements under policies such as SERPT. However, this insight is only made when the job processing times have relatively low variance. As discussed below, when job processing times have high variance, underestimating even a single very large job can severely affect the smaller jobs' sojourn times.

The work in (4) has the optimistic viewpoint that it is possible to obtain improved performance by utilizing processing time estimates in a simple manner. The more pessimistic view is that when job processing times are estimated, estimation errors naturally arise, and they can degrade a scheduling policy's performance, if the policy was designed to exploit exact knowledge of job processing times (5). The SERPT policy may have poor performance when the job processing times have high variance and large jobs are underestimated. Consider a situation where a job with a processing time of 1000 enters the system and is underestimated by 10%. The moment the job has been processed for 900 units (its estimated processing time), the server assumes that this job's estimated remaining processing time is zero, and until it completes, the job will block the jobs already in the queue as well as any new arrivals. This situation becomes more severe when both the actual job processing time and the level of underestimation increase. However, when the job processing times are generated from lower variance distributions, the underestimation of large jobs will not cause severe performance degradation (6).

The Shortest Estimated Processing Time (SEPT) policy is a version of the Shortest Processing Time (SPT) policy that skips updating the estimated remaining processing times and prioritizes jobs based only on their estimated processing times. Experimental results show that SEPT has impressive performance in the presence of estimated job processing times, as well as being easier to implement than SERPT (7).

In this chapter, we will discuss the problem of single-server scheduling when only

estimates of the job processing times are available. In Section 2.2, we discuss the existing literature for scheduling policies that handle inexact job processing time information. Most of the existing literature analyzes and introduces size-based policies when the estimation error is relatively small, restricting applicability of the results. Furthermore, many simulation-based examinations only consider certain workload classes and are not validated over a range of job processing times and estimation error distributions. We propose a scheduling policy that exhibits desirable performance over a wide range of job processing time distributions, estimation error distributions, and workloads.

The Gittins' Index policy (8), a dynamic priority-based policy, is optimal in minimizing the MST in an M/G/1 queue (9). When there are job processing time estimates, the Gittins' Index policy utilizes information about job estimated processing time, and the job processing time and estimation error distributions to decide which job should be processed next. The assumption of knowing these distributions before scheduling may be problematic in real environments. Furthermore, scheduling jobs using the Gittins' Index policy introduces computational overhead that may be prohibitive. While there are significant barriers to implementing the Gittins' Index policy, our proposed policy is motivated by the form of the Gittins' Index policy.

We make the following contributions: While the SEPT policy performs well in the presence of estimated job processing times (7), we first introduce a heuristic that combines the merits of SERPT and SEPT. Secondly, we specify the Gittins' Index policy given multiplicative estimation errors and restricted to knowing only the estimation error distribution. We show that our proposed policy, which we call the Size Estimate Hedging (SEH) policy, has performance close to the Gittins' Index policy. Similar to SERPT and SEPT, the SEH policy only uses the job processing time estimates to prioritize the jobs. Finally, we provide

numerical results obtained by running a wide range of simulations for both synthetic and real workloads. The key observations suggest that SEH outperforms SERPT except in scenarios where the job processing time variance is extremely low. SEH outperforms SEPT whether the variance of the job processing times is high or low. With the presence of better estimated processing times in the system (low variance in the estimation errors), SEH outperforms SEPT and has performance close to the optimal policy (SRPT) if the estimation errors are removed. On the other hand, we observe that when the estimation errors have high variance, there is little value in using the estimated processing times. We also notice that the system load does not significantly affect the relative performance of the policies under evaluation. The SEH policy treats underestimated and overestimated jobs fairly, in contrast with other policies that tend to favor only one class of jobs. Even though the policy does not directly consider fairness between the underestimated and the overestimated jobs, it results in a more equal treatment of the underestimated and the overestimated jobs by reducing the priority of the underestimated jobs when the underestimation is certain. When the job processing time variance is high, the SEH and SEPT policies obtain a near-optimal mean slowdown value of 1, indicating that underestimated large jobs do not delay small jobs. In terms of mean slowdown, SEH outperforms SEPT across all levels of job processing time variance.

The rest of the chapter is organized as follows. Section 2.2 presents the existing literature in scheduling single-server queues with estimated job processing times. Section 2.3 defines our SEH policy and discusses its relationship to a Gittins' Index approach. Our simulation experiments are described in detail in Section 2.4. We provide the results of our simulations in Section 2.5 and conclude and discuss future directions in Section 2.6.

2.2 Related Work

Scheduling policies and their performance evaluation in a preemptive M/G/1 queue have been a subject of interest for some time. Size-based policies are known to perform better than size-oblivious policies with respect to sojourn times. In fact, the SRPT policy is optimal in minimizing the MST (2). However, size-based policies have a considerable disadvantage: When the exact processing times are not known to the system before scheduling, which is often the case in practical settings, their performance may significantly degrade. Dell'Amico et al. (10) study the performance of SRPT with estimated job processing times and demonstrate the consequences of job processing time underestimations under different settings. Studies in Harchol-Balter et al. (11) and Chang et al. (12) discuss the effect of inexact processing time information in size-based policies for web servers and MapReduce systems, respectively. Our chapter assumes that the processing time is not available to the scheduler until the job is fully processed, but that processing time estimations are available. The related literature for this setting is reviewed in the following paragraph.

Lu et al. (5) were the first to study this setting. They show that size-based policies only benefit the performance when the correlation between a job's real and estimated processing time is high. The results in Wierman and Nuyens (13), Bender et al. (14), and Becchetti et al. (15) are obtained by making assumptions that may be problematic in practice. A strict upper bound on the estimation error is assumed in (13). On the other hand, (15) and (14) define specific job processing time classes and schedule the jobs based on their processing time class, which can be problematic for very small or very large jobs. This setting is also known as semi-clairvoyant scheduling. In this work, we do not assume any bounds on the estimation error or assign jobs to particular processing time classes. Consistent with this body of work, we do find that SEH is not recommended for systems with large estimation

error variance. However, we do find that it performs well for levels of estimation error variance that are typically found in practice.

When the job processing time distribution is available, the Gittins' Index policy (8) assigns a score to each job based on the processing time it has received so far, and the scheduler chooses the job with the highest score to process at each point in time. This policy is proven to be optimal for minimizing the MST in a single-server queue when the job processing time distribution is known (9). This policy is specified in the next section.

2.3 Size Estimate Hedging: A Simple Dynamic Priority Scheduling Policy

2.3.1 Model

Consider an M/G/1 queue where preemption is allowed and we are interested in minimizing the MST. We assume that a job's processing time is not known upon arrival; however, an estimated processing time is provided to the scheduler. We concentrate on a multiplicative error model where the error distribution is independent of the job processing time distribution. The estimated processing time $\hat{S}$ of a job is defined as $\hat{S} = SX$ where S is the job processing time and X is the job processing time estimation error. We assume that the value of $\hat{S}$ is known upon each job's arrival and is denoted by $\hat{s}$. The choice of a multiplicative error model results in having an absolute error proportional to the job processing time S, thus avoiding situations where the estimation errors tend to be worse for small jobs than for large jobs. Furthermore, Dell'Amico et al. (10) and Pastorelli et al. (16) suggest that a multiplicative error model is a better reflection of reality. To define our scheduling policies, we also require the notion of a quantum of service. The job with the highest priority is

processed for a quantum of service Δ until either it completes or a new job arrives. At that point, priorities are recomputed.

2.3.2 Gittins' Index Approach

The Gittins' Index Policy is an appropriate technique for determining scheduling policies when the job processing time and estimation error distributions are known. For a waiting job i, an index $G(a_i)$ is calculated, where a_i is the elapsed processing time. At each time epoch, the Gittins' Index policy processes the job with the highest index $G(a)$ among all of the present waiting jobs (8). The Gittins' rule takes the job's elapsed processing time into account and calculates the optimal quantum of service $\Delta^*(a)$ that it should receive.

The associated efficiency function $J(a,\Delta), a, \Delta \geq 0$ of a job with processing time S, elapsed processing time a and quantum of service Δ is defined as

$$J(a,\Delta) = \frac{P(S-a \leq \Delta | S > a)}{E[\min\{S-a,\Delta\}|S > a]}. \tag{2.3.1}$$

The numerator is the probability that the job will be completed within a quantum of service Δ, and the denominator is the expected remaining processing time a job with elapsed processing time a and quantum of service Δ will require to be completed.

The server (preemptively) processes the job with the highest index at each decision epoch. Decisions are made when (i) a new job arrives to the queue, (ii) the current job under processing completes, or (iii) the current job receives its optimal quantum of service and does not complete. If there are multiple jobs that have the same highest index and all have zero optimal quanta of service, the processor will be shared among them as long as this situation does not change. If there is only one job with the highest index and zero optimal quantum of service, its index should be updated throughout its processing (9).

Although the Gittins' Index policy is optimal in terms of minimizing the mean sojourn time in an $M/G/1$ queue (9), the assumption of knowing the job size and estimation error distributions might not always be practical to make. Furthermore, forming the Gittins' Index policy's efficiency function has significant computational overhead. As a result, this policy may be a problematic choice for real environments where the scheduling speed is important. However, examining the form of optimal policies has helped us in the construction of a simple heuristic. In particular, the notion of defining a policy in terms of an index allows us to make precise our notion of combining the relative merits of SRPT and SEPT.

2.3.3 Motivation

When a job enters the system under SERPT, there is no basis on which to assume that the estimated processing time, $\hat{s}$, is incorrect. However, when the elapsed processing time reaches $\hat{s}$, we are certain that the job processing time has been underestimated. In addition, Dell'Amico et al. (7) show that SEPT performs well when dealing with estimated processing times and in the presence of estimation errors, in particular severe underestimates. So, we would like to combine these two policies. A convenient way to do this is to introduce a Gittins'-like score function, where a higher score indicates a higher priority. We will be aggressive and use the score function for SERPT until the point that we know a job is underestimated and then freeze the score, which is similar to what SEPT's constant score function does (see (2.3.4) below). In this way, instead of switching to SEPT's score function, we would like to give credit for the jobs' cumulative elapsed processing times.

The score functions for SRPT, SERPT, and SEPT are provided in (2.3.2), (2.3.3), and

(2.3.4), respectively.

$$G(a,s) = \frac{1}{s-a}, \tag{2.3.2}$$

$$G(a,\hat{s}) = \begin{cases} \frac{1}{\hat{s}-a}, & \hat{s} > a, \\ \infty, & \hat{s} \leq a, \end{cases} \tag{2.3.3}$$

$$G(a,\hat{s}) = \frac{1}{\hat{s}}. \tag{2.3.4}$$

We note that (2.3.2) and (2.3.3) have an increasing score function, and (2.3.4) always assigns a constant score for a particular job.

2.3.4 The SEH Policy

Combining the score functions for SERPT and SEPT, we now define our policy. As discussed in the previous section, we would like to transition between SERPT when we cannot determine if a job processing time is underestimated to a fixed priority like SEPT when it is determined that underestimation has occurred. One consequence of using this policy is that any underestimated small job can still receive a "high" score and be processed, while underestimated large jobs will have a much lower score and do not interfere, even with underestimated small jobs. Furthermore, not needing to know the job processing time and estimation error distribution, the SEH Policy does not have much overhead. Thus, it can schedule the jobs at a speed comparable to the SEPT policy.

We introduce the score function of our SEH policy as

$$G(a,\hat{s}) = \begin{cases} \frac{1}{\hat{s}-a(1-\frac{a}{2\hat{s}})}, & 0 \le a < \hat{s}, \\ \frac{2}{\hat{s}}, & a \ge \hat{s}, \end{cases} \tag{2.3.5}$$

where the scheduling decisions are only made at arrivals and departures.

With the score function in (2.3.5), a job's score will increase up to the point that it receives processing equal to its estimated processing time and then receives a constant score of $\frac{2}{\hat{s}}$ until it completes. The choice of 2 was made after some experimentation, it would be worthwhile to explore the sensitivity of the performance to this choice.

2.3.5 Gittins' Index vs. SEH

In this section, we show that the form of our policy is consistent with the Gittins' index in the setting that we only know the error estimate distribution. In particular, we have no a priori or learned knowledge of the processing time distribution.

With our estimation model in mind and with the additional knowledge of the estimate $\hat{s}$, (2.3.1) can be rewritten as

$$J(a,\Delta,\hat{s}) = \frac{P(\frac{\hat{s}}{X} - a \le \Delta | \frac{\hat{s}}{X} > a)}{E[\min\{\frac{\hat{s}}{X} - a, \Delta\} | \frac{\hat{s}}{X} > a]} \tag{2.3.6}$$

where the numerator can be evaluated using the definition of conditional probability:

$$P(\frac{\hat{s}}{X} \le a + \Delta | \frac{\hat{s}}{X} > a) = \frac{P(\frac{\hat{s}}{a+\Delta} \le X < \frac{\hat{s}}{a})}{P(X < \frac{\hat{s}}{a})} \tag{2.3.7}$$

The density and distribution of the estimation error are denoted by f_X and F_X, respectively. Suppose that the lower and upper limits on the estimation error distribution are l and

u, respectively (l may be zero and u may be ∞). Then, by considering cases, the RHS of (2.3.7) (and hence the numerator of (2.3.6)) evaluates to:

$$P(\frac{\hat{s}}{X}-a\leq\Delta|\frac{\hat{s}}{X}>a)=\begin{cases}1, & \frac{\hat{s}}{a+\Delta}\leq l<\frac{\hat{s}}{a}<u,\\ 1-\frac{F_X(\frac{\hat{s}}{a+\Delta})}{F_X(\frac{\hat{s}}{a})}, & l<\frac{\hat{s}}{a+\Delta}<\frac{\hat{s}}{a}<u,\\ 1, & \frac{\hat{s}}{a+\Delta}\leq l<u<\frac{\hat{s}}{a},\\ 1-F_X(\frac{\hat{s}}{a+\Delta}), & l<\frac{\hat{s}}{a+\Delta}<u\leq\frac{\hat{s}}{a}.\end{cases}$$

To calculate the denominator in (2.3.6), we first compute the required conditional probability density function as

$$f_{X|X<\frac{\hat{s}}{a}}(x)=\frac{f_X(x)}{\int_l^{\frac{\hat{s}}{a}}f_X(y)dy}=\begin{cases}\frac{f_X(x)}{\int_l^{\frac{\hat{s}}{a}}f_X(y)dy}, & \frac{\hat{s}}{a}<u,\\ f_X(x), & otherwise.\end{cases}$$

The denominator in (2.3.6) can then be written as

$$E[\min\{\frac{\hat{s}}{X}-a,\Delta\}|\frac{\hat{s}}{X}>a]=\begin{cases}\int_l^{\frac{\hat{s}}{a}}(\hat{s}-ax)\frac{f_X(x)}{\int_l^{\frac{\hat{s}}{a}}f_X(y)dy}dx, & \frac{\hat{s}}{a+\Delta}\leq l<\frac{\hat{s}}{a}<u,\\ \int_l^{\frac{\hat{s}}{a+\Delta}}ax\frac{f_X(x)}{\int_l^{\frac{\hat{s}}{a}}f_X(y)dy}dx+\int_{\frac{\hat{s}}{a+\Delta}}^{\frac{\hat{s}}{a}}(\hat{s}-ax)\frac{f_X(x)}{\int_l^{\frac{\hat{s}}{a}}f_X(y)dy}dx, & l<\frac{\hat{s}}{a+\Delta}<\frac{\hat{s}}{a}<u,\\ \int_l^{u}(\hat{s}-ax)f_X(x)dx, & \frac{\hat{s}}{a+\Delta}\leq l<u<\frac{\hat{s}}{a},\\ \int_l^{\frac{\hat{s}}{a+\Delta}}x\Delta f_X(x)dx+\int_{\frac{\hat{s}}{a+\Delta}}^{u}(\hat{s}-ax)f_X(x)dx, & l<\frac{\hat{s}}{a+\Delta}<u\leq\frac{\hat{s}}{a}.\end{cases}$$

Considering

$$E[X|X < \frac{\hat{s}}{a}] = \frac{\int_l^{\frac{\hat{s}}{a}} x f_X(x)dx}{P(X < \frac{\hat{s}}{a})},$$

and

$$E[X] = \int_l^u x f_X(x)dx,$$

(2.3.6) can be rewritten as

$$J(a,\Delta,\hat{s}) = \begin{cases} \frac{1}{\hat{s}-aE[X|X\leq\frac{\hat{s}}{a}]}, & \frac{\hat{s}}{a+\Delta} \leq l < \frac{\hat{s}}{a} < u, \\ \frac{P(\frac{\hat{s}}{a+\Delta}\leq X\leq\frac{\hat{s}}{a})}{\Delta E[X|X\leq\frac{\hat{s}}{a+\Delta}]P(X\leq\frac{\hat{s}}{a+\Delta})+P(\frac{\hat{s}}{a+\Delta}\leq X\leq\frac{\hat{s}}{a})(\hat{s}-aE[X|\frac{\hat{s}}{a+\Delta}\leq X\leq\frac{\hat{s}}{a}])}, & l < \frac{\hat{s}}{a+\Delta} < \frac{\hat{s}}{a} < u, \\ \frac{1}{\hat{s}-aE[X]}, & \frac{\hat{s}}{a+\Delta} \leq l < u < \frac{\hat{s}}{a}, \\ \frac{1-P(X\leq\frac{\hat{s}}{a+\Delta})}{\Delta E[X|X\leq\frac{\hat{s}}{a+\Delta}]P(X\leq\frac{\hat{s}}{a+\Delta})+P(X\geq\frac{\hat{s}}{a+\Delta})(\hat{s}-aE[X|X\geq\frac{\hat{s}}{a+\Delta}])}, & l < \frac{\hat{s}}{a+\Delta} < u \leq \frac{\hat{s}}{a}. \end{cases} \tag{2.3.8}$$

The Gittins' index $G(a,\hat{s}), a \geq 0$, is defined by

$$G(a,\hat{s}) = \sup_{\Delta\geq 0} J(a,\Delta,\hat{s}).$$

The optimal quantum of service is denoted as

$$\Delta^*(a,\hat{s}) = \sup\{\Delta \geq 0 | G(a,\hat{s}) = J(a,\Delta,\hat{s})\}.$$

At $\Delta = \frac{\hat{s}}{l} - a$, the first and third case in (2.3.8) are equal to the second and fourth case, respectively, and $J(a,\Delta,\hat{s})$ is maximized.

The Gittins' index can then be written as

$$G(a,\hat{s}) = \begin{cases} \frac{1}{\hat{s}-aE[X|X\leq\frac{\hat{s}}{a}]}, & \frac{\hat{s}}{a} < u, \\ \frac{1}{\hat{s}-aE[X]}, & otherwise, \end{cases} \tag{2.3.9}$$

where $\Delta^* = \frac{\hat{s}}{l} - a$. For instance, the Gittins' index for a $Log-N(\mu,\sigma^2)$ error distribution is

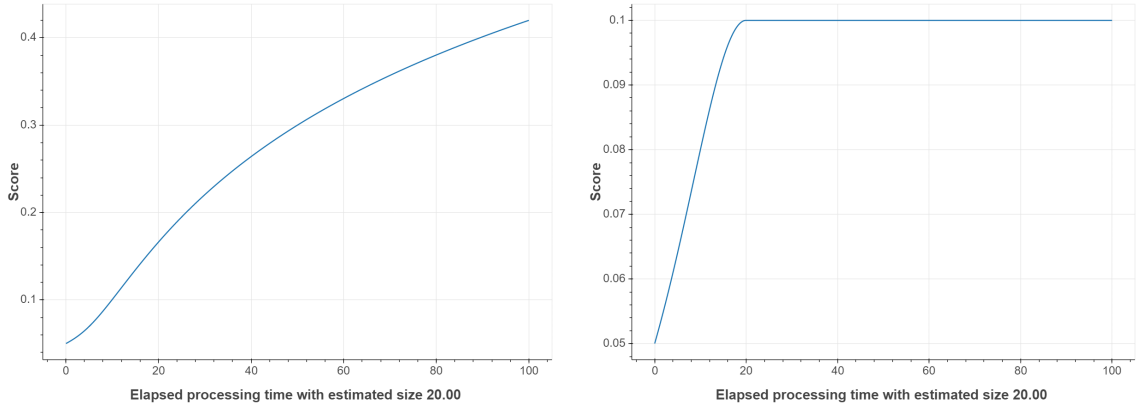
$$G(a,\hat{s}) = \frac{1}{\hat{s} - ae^{\mu+g(a,\hat{s})}}, \tag{2.3.10}$$

where

$$g(a,\hat{s}) = \frac{\sigma^2\phi[\frac{\ln(\frac{\hat{s}}{a})-\mu-\sigma^2}{\sigma}]}{2\phi[\frac{\ln(\frac{\hat{s}}{a})-\mu}{\sigma}]},$$

and ϕ is the cumulative distribution function of the $Log-N(0,\sigma^2)$ distribution. Note that for the Log-Normal distribution as the job processing time error distribution, the second case in (2.3.9) cannot happen. For the remainder of the chapter, we will refer to this policy as the Gittins' Index policy. We recognize that this is a slight abuse of terminology, as we are ignoring the job processing time distribution.

Taking the score in (2.3.10) into account, for any job with an estimated processing time $\hat{s}$, the score calculated with the Gittins' Index policy continuously increases until the job completes. Fig. 2.1a shows this score for a job with an estimated processing time of 20 and an estimation error generated from a $Log-N(0,\sigma^2)$ distribution as a function of its elapsed processing time. We observe that for larger values of elapsed processing time, the slope of the score is decreasing. Fig. 2.1b shows the score calculated with the SEH policy for a job with an estimated processing time of 20 as a function of its elapsed processing time. The score shown in Fig. 2.1a is consistent with the score function having decreasing slope at



(a) calculated with the Gittins' Index policy (b) calculated with SEH

Figure 2.1: Job score as a function of the elapsed processing time

some point beyond the point at which the elapsed processing time reaches the estimated processing time, as in Fig. 2.1b. Of course, the change in slope for SEH is more severe, but we will see in our numerical experiments that the performance of the two policies is quite close. SEH has less computational overhead and more importantly, does not require knowledge of the estimation error distribution.

2.4 Evaluation Methodology

2.4.1 Policies Under Evaluation

In this section, we introduce the size-based scheduling policies considered for evaluation. As our baseline policy, we consider the SRPT policy when the exact job processing times, given by s, are known before scheduling. The SRPT policy is an "ideal" policy since it assumes that there are no errors in estimating the processing time.

- **SERPT policy** — The SERPT policy is a version of SRPT that uses the estimates of job processing times as if they were the true processing times.

- **SEPT policy** — The SPT policy skips the SRPT policy's updating of remaining processing times and only schedules jobs based on their estimated processing time.
- **SEH and Gittins' Index policy** — Our proposed SEH policy and the Gittins' Index policy are explained in detail in Section 2.3.4 and Section 2.3.5, respectively.

All these policies fit into the "scoring" framework, and they assign scores to each job and process the jobs in the queue in the descending order of their scores. Moreover, preemption is allowed, and a newly-arrived job can preempt the current job if it has a higher score. The score functions in (2.3.2), (2.3.3), (2.3.4), (2.3.5), and (2.3.9) show how we calculate the scores for the SRPT, SERPT, SEPT, SEH, and Gittins' Index policy, respectively.

2.4.2 Performance Metrics

We evaluate the policies defined in Section 2.4.1 with respect to two performance metrics: MST and Mean Slowdown. When the job processing times have large variance, the sojourn times for small jobs and large jobs differ significantly. Thus, we use the per job slowdown, the ratio between a job's sojourn time and its processing time (17).

2.4.3 Simulation Parameters

We would like to evaluate the policies over a wide range of job processing time and error distributions. To generate this range of distributions, we fix the form of the distribution and vary the parameters. We use the same settings that Dell'Amico et al. (10) use in their work. Table 2.1 provides the default parameter values that we use in our simulation study. We now provide details of our simulation model. Note that our policy fits into the SOAP framework of Scully et al. (18), however as we are also evaluating mean slowdown, we

Table 2.1: Parameter Settings

Parameter	Definition	Default
# jobs	the number of departed jobs	$10,000$
k	shape for Weibull job processing time distribution	0.25
σ	σ in the Log-Normal error distribution	0.5
ρ	system load	0.9

chose simulation for evaluation.

Job Processing Time Distribution — We consider an $M/G/1$ queue where the processing time is generated according to a Weibull distribution. This allows us to model high variance processing time distributions, which better reflect the reality of computer systems (see (19), (20) for example). In general, the choice of a Weibull distribution gives us the flexibility to model a range of scenarios. The shape parameter k in the Weibull distribution allows us to evaluate both high variance (smaller k) and low variance (larger k) processing time distributions.

Considering that the job processing time distribution plays a significant role in the scheduling policies' performance and size-based policies show different behaviors with high variance job processing time distributions, we choose $k = 0.25$ as our default shape for the Weibull job processing time distribution. With this choice for k, the scheduling policies' performance is highly influenced by a few very large jobs that constitute a substantial percentage of the system's overall workload. We vary k between 0.25 and 2, considering specific values of 0.25, 0.375, 0.5, 0.75, 1, and 2. We show that the SEH policy performs best in the presence of high variance job processing time distributions.

Job Processing Time Error Distribution — We have chosen the Log-Normal distribution as our error distribution so that a job has an equal probability of being overestimated or underestimated. The Gittins' index for this estimation error distribution is shown in (2.3.10). The σ parameter controls the correlation between the actual and estimated processing time, as well as the estimation error variance. By increasing the σ value, the correlation coefficient becomes smaller, and the estimation error variance increases, resulting in the occurrence of more large underestimations/overestimations (more imprecise processing times). We choose $\sigma = 0.5$ as the default value that corresponds to a median relative error factor of 1.40. We vary σ between 0.25 and 1 with specific values of 0.25, 0.375, 0.5, 0.75, and 1 to better illustrate the effect of σ on the evaluated performance.

System Load — Following Lu et al. (5), we consider $\rho = 0.9$ as the default load value and vary ρ between 0.5 (lightly loaded) and 0.95 (heavily loaded) with increments of 0.05 and an additional system load of 0.99.

Number of Jobs — The number of jobs in each simulation run is $10,000$ and a simulation run ends when the first $10,000$ jobs that arrived to the system are completed. We fix the confidence level at 95%, and for each simulation setting, we continue to perform simulation runs until the width of the confidence interval is within 5% of the estimated value. For low variance processing time distributions (larger k), 30 simulation runs suffice; however, more simulation runs are required for high variance processing time distributions (smaller k).

2.5 Simulation Results

In this section, we evaluate the performance of the policies in Section 2.4.1 by running experiments on both synthetic and real workloads. We run different simulations by generating synthetic workloads based on different job processing time and error parameters and we analyze these parameters' effect on the performance of each of the policies.

For evaluating our results in practical environments, we consider a real trace from a Facebook Hadoop cluster in 2010 (21) and show that the policies' performance is consistent with the results we obtained with synthetic workloads. The key observations, validated both on synthetic and real workloads, are highlighted as follows:

- The Gittins' Index policy outperforms SERPT for all the evaluated values of k and σ. We show the same observation with our proposed SEH policy except for values of k that correspond to very low job processing time variance.
- The Gittins' Index and SEH policies outperform SEPT with lower values of σ (better estimated processing times) and have an MST near the optimal MST obtained without any estimation errors.
- SEH performs well in reducing both the MST of overestimated jobs and underestimated jobs.
- The load parameter does not have a significant effect on the relative values of the MST obtained with the evaluated policies.
- The Gittins' Index, SEH and SEPT policies have a near-optimal mean slowdown of 1 when the estimated processing times have high variance.

- The SEH performs best across all values of k in terms of minimizing the mean slowdown.

In what follows, we discuss the numerical results and how they support these key observations.

Synthetic Workloads — We first note that the job processing time k parameter and the estimation error σ parameter have the greatest impact on the policies' performance. Thus, we focus on varying these parameters. We show that the Gittins' Index policy outperforms SERPT across all evaluated values of k and σ and our SEH policy outperforms SERPT except for the values of k and σ that correspond to distributions with extremely low variance. For the scenarios where we do not state the parameter values explicitly, the parameters in Table 2.1 (see Section 2.4.3) are considered.

Fig. 2.2 captures the impact of job processing time variance and displays the MST of the Gittins' Index, SEH, SERPT, and SEPT policies normalized against the MST obtained with SRPT with σ having the default value of 0.5. We observe that for a high variance job processing time distribution ($k = 0.25$), SERPT performs very poorly compared to the other policies due to the presence of large, underestimated jobs. We note that the SERPT policy performs well if the variance of the processing times is sufficiently low. Based on Fig. 2.2, we notice that the gap between SEPT and the Gittins' Index policy grows slightly when the job processing time variance is lower. The gap between SEH and the Gittins' Index policy also grows but not to the same degree as SEPT. For $k > 0.75$, the performance of the Gittins' Index policy, SEH, and SERPT are quite close. In fact, we observe that our SEH policy performs very close to the Gittins' Index policy across all values of k. Furthermore, we notice that as the variance in processing times gets smaller, the gap between what is

achievable by the policy under evaluation and what is achievable if there were no errors is larger than for the high variance scenarios.

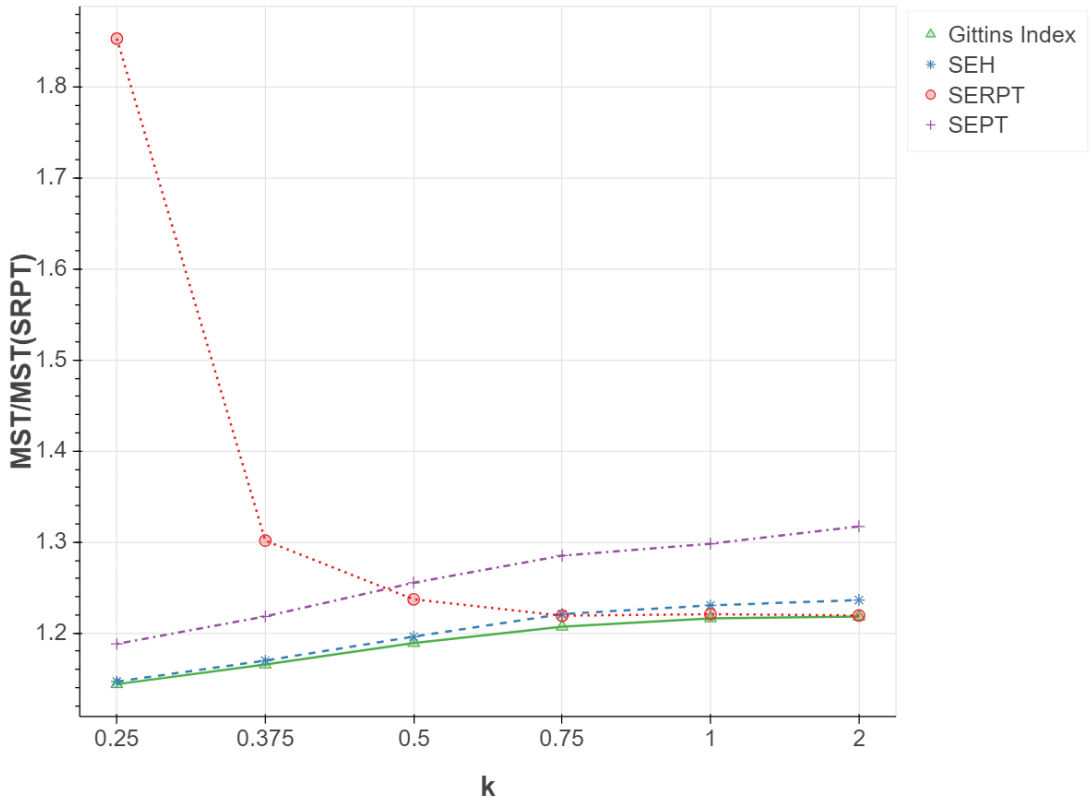


Figure 2.2: Impact of k on the MST

The shape parameter k affects the job processing time variance and the scheduling policies' performance the most, especially when the job processing time distribution has high variance. We can be optimistic about using estimates if the variance is low, but we have to be careful in choosing the scheduling policy if the job processing time variance is high. The literature focuses on high variance workloads, and we will continue evaluating the policies on such workloads. In Fig. 2.3, we display the normalized MST of the policies against the MST of the SRPT policy under varying σ, $\rho = 0.9$, and the default $k = 0.25$. We notice that the Gittins' Index, SEH, and SEPT policies are relatively insensitive to the

σ value, while the gap between these three policies and SERPT increases with increasing σ. In fact, the Gittins' Index and SEH policies outperform SEPT with $\sigma \leq 0.5$ and have an MST near the optimal MST obtained without any estimation errors. We conclude that the impact of the Gittins' Index policy and SEH becomes more prominent when the estimates improve.

In Fig. 2.3, we observe that while choosing a more aggressive policy like the Gittins' Index and SEH policies is a good choice under lower values of σ, SEPT is preferred when $\sigma = 1$. The reason is that lower values of k (here, $k = 0.25$), cause more large jobs in the system. Furthermore, for values of $\sigma \geq 1$, the estimation errors have high variance and thus the estimated processing times can be very imprecise. We notice that both SEH and the Gittins' Index policy suffer from a slight promotion of severely underestimated jobs that leads to temporary blockage for the other jobs. What has happened in this case is that the estimates of the processing times have degraded to the point that they are not useful. In particular, one should instead base scheduling decisions on the processing time distribution, so for example in scenarios with high variance in both processing times and estimation errors, a policy which ignores the estimates, such as Least Attained Service (LAS) would be warranted. The LAS scheduling policy (22), also known as Shortest Elapsed Time (23) and Foreground-Background (24), preemptively prioritizes the job(s) that have been processed the least. If more than one job has received the least amount of processing time, the jobs will share the processor in a processor-sharing mode. Analytic results in (25), (26) show that LAS minimizes MST when the job processing time distribution has a decreasing hazard rate and there are no processing time estimates available.

These observations are consistent with the results in Table 2.2 which considers the same settings as in Fig. 2.3 when $\sigma = 1$ and $\sigma = 2$. SERPT has poor performance compared

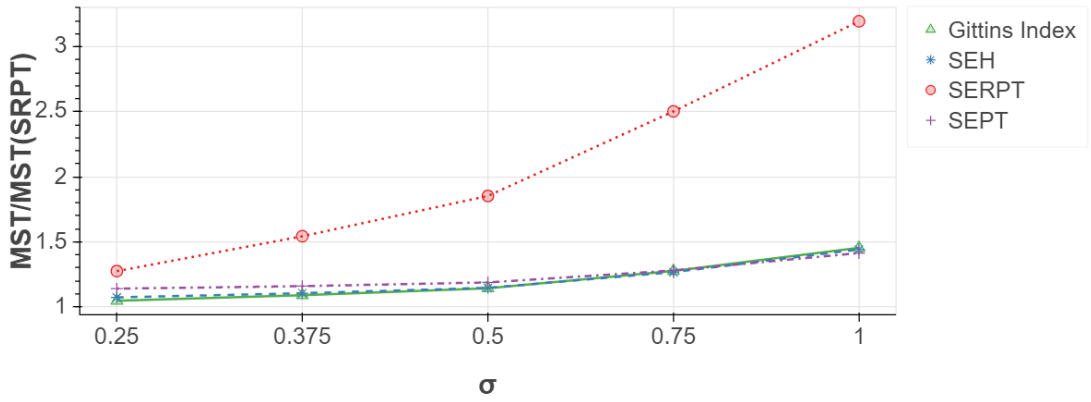


Figure 2.3: Impact of σ on the MST

Table 2.2: Policies evaluation under $\sigma = 1$ and $\sigma = 2$

	$\sigma = 1$		$\sigma = 2$	
Policy	MST/ MST(SRPT)	Mean Slowdown	MST/ MST(SRPT)	Mean Slowdown
Gittins' Index	1.45	1.26	2.68	6.78
SEH	1.44	1.22	2.71	6.87
SEPT	1.41	1.16	2.54	4.71
LAS	1.81	1.27	1.81	1.27
SRPT	1	1.06	1	1.06

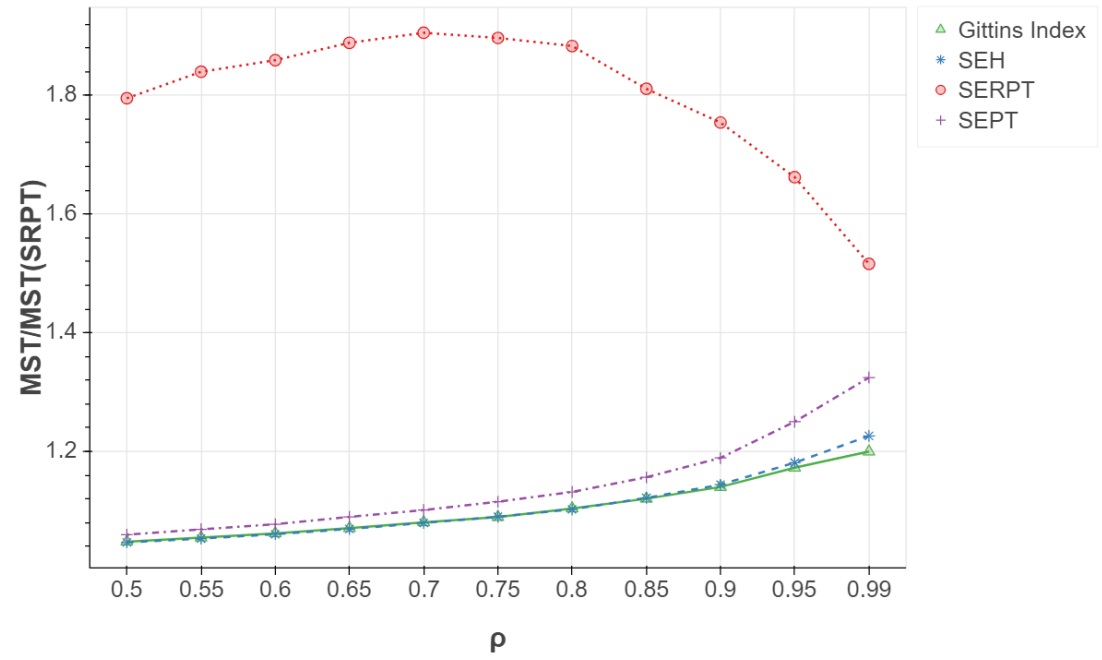
the other policies under $\sigma \geq 1$ and thus is not included. Pastorelli et al. (16) show that lower values of σ ($\sigma < 1$) are what one sees in practice. It would be interesting to look at the optimal Gittins' Index policy that includes both the job processing time and estimation error distributions, as it would capture this effect. Although doing so can help develop policies that are effective even at high values of σ, deriving the Gittins' index would be quite complicated with this extra condition, but it could give insight into designing simpler policies.

Fig. 2.4a, Fig. 2.4b, and Fig. 2.4c show the result of simulations with the default values in Table 2.1 and varying the system load between 0.5 and 0.99 for all jobs, only the overestimated jobs, and only the underestimated jobs, respectively. If we concentrate only

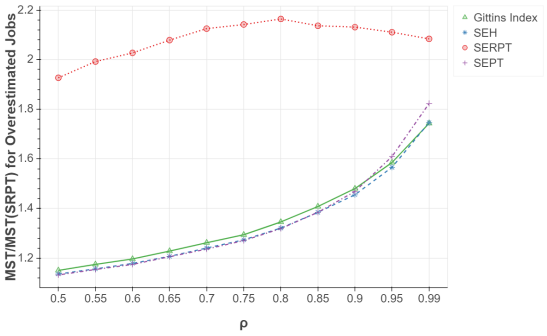
on one class of jobs (overestimated or underestimated), the policy that minimizes the MST the most can be different. We observe that the Gittins' Index and SEH policies perform best in minimizing the overall MST given different system loads. The Gittins' Index policy performs best in reducing the MST of underestimated jobs and the SEH policy has desirable performance in reducing the MST of all jobs, the overestimated jobs, and the underestimated jobs. Fig. 2.4a shows that the load parameter does not have a significant effect on the MST since the ratio between the MST of each policy and the MST of SRPT remains almost unchanged.

The mean slowdown is the other metric we consider to evaluate the performance of the policies. High values of mean slowdown indicate that some jobs spend a disproportionate amount of time waiting. In Fig. 2.5, we show the mean slowdown for different values of k with $\rho = 0.9$ and a σ value of 0.5. The mean slowdown of SERPT is not included since it is several orders of magnitude higher for $k \leq 0.5$. We see that the Gittins' Index, SEH, and SEPT policies have similar performance. All policies have a near-optimal mean slowdown of 1 for high variance job processing time distributions (smaller k). The reason is that the very small jobs (that make up the majority of the jobs) are processed the moment they enter the system, and no large job blocks them. We also observe that SEH performs best across all values of k in terms of minimizing the mean slowdown.

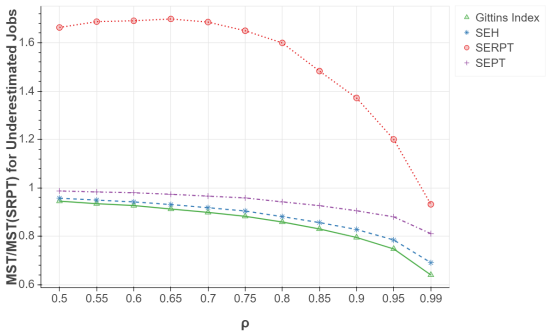
We conclude our experiments with synthetic workloads by indicating that the Gittins' Index and SEH policies perform better than SERPT under different parameter settings. The only exception is extreme situations like the low variance job processing time distributions (larger k) where SERPT outperforms SEH and works analogously to the Gittins' Index policy.



(a) All jobs



(b) Overestimated jobs



(c) Underestimated jobs

Figure 2.4: Impact of ρ on the MST

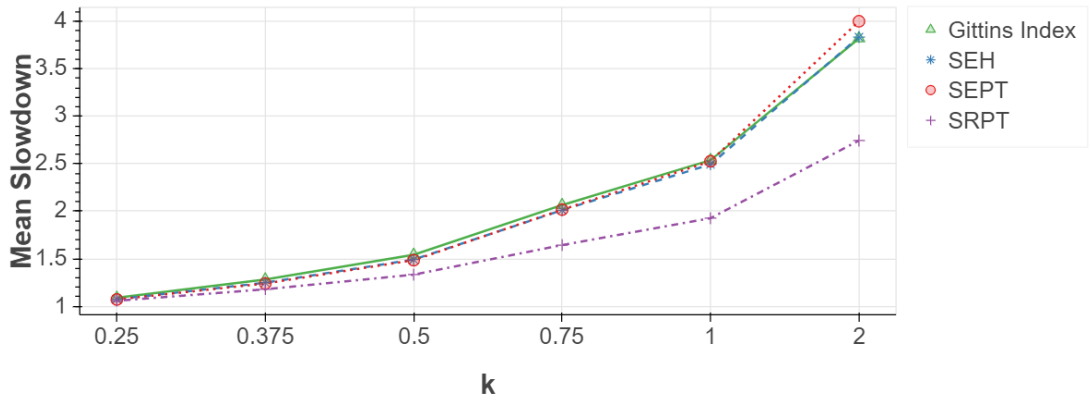


Figure 2.5: Impact of k on the mean slowdown

Real Workloads — We consider a Facebook Hadoop cluster trace from 2010 (21) and show that the results with this workload look very similar to those with synthetic workloads generated with $k = 0.25$. The trace consists of $24,443$ jobs. We assume each job's processing time is the sum of its input, intermediate output, and final output bytes. The job processing times of this workload have high variance, and thus, we run hundreds of simulations to reach the desired confidence interval (as described in Section 2.4.3). We vary the error estimation distribution's σ parameter to evaluate different scenarios of estimated processing time precision. To maintain the default settings in Table 2.1, we define the processing speed in bytes per second. The arrival rate λ is chosen to yield the desired $\rho = 0.9$. A simulation run ends when the last job in the workload arrives at the system and we calculate the MST of the jobs that are fully processed among the first $10,000$ jobs that entered the system. Fig. 2.6 shows the MST normalized against the optimal MST obtained with SRPT with varying σ between 0.25 and 1. We observe that the Gittins' Index and SEH policies perform best across all values of σ.

In Fig. 2.7, we display the mean slowdown obtained with the policies under evaluation. Similar to Fig. 2.5, we have not included the mean slowdown of SERPT since it is several

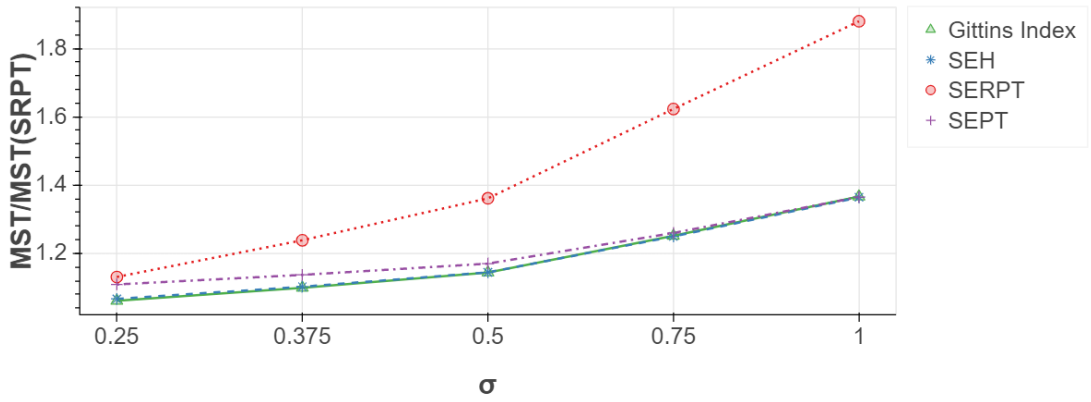


Figure 2.6: MST of the Facebook Hadoop workload

orders of magnitude higher. We observe that for $\sigma \leq 0.5$, where the estimates are better, the SEH policy has lower mean slowdown than the Gittins' Index and SEPT policies, however, SEPT starts to outperform the Gittins' Index and SEH policies when σ increases, consistent with our observations for synthetic workloads.

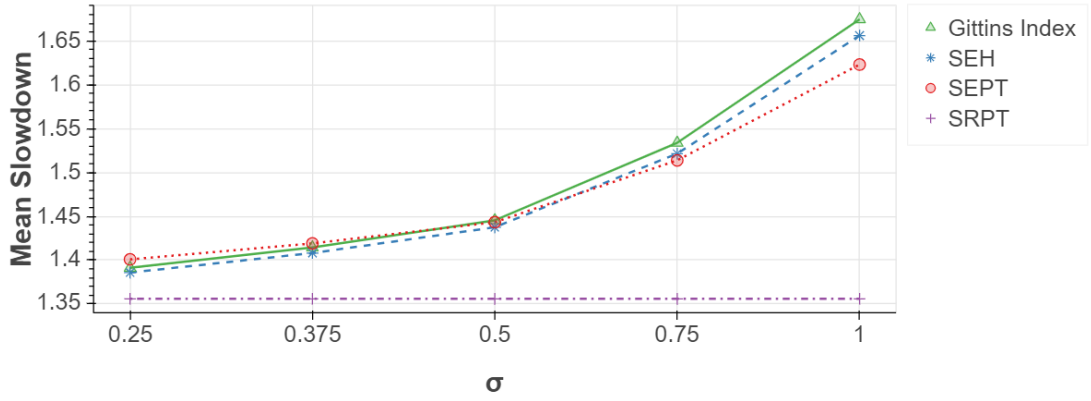


Figure 2.7: Mean slowdown of the Facebook Hadoop workload

2.6 Conclusion and Future Work

The SRPT policy, which is optimal for scheduling in single-server systems, may have problematic performance when job processing times are estimated. This work has considered the problem of scheduling with the presence of job processing time estimates. A multiplicative error model is used to produce estimation errors proportional to the job processing times. We have introduced a novel heuristic that combines the merits of SERPT and SEPT and requires minimal calculation overhead and no information about the job processing time and estimation error distributions. We have shown that this policy is consistent with a Gittins'-like view of the problem. Our numerical results demonstrate that the SEH policy has desirable performance in minimizing both the MST and mean slowdown of the system when there is low variance in the estimation error distribution. It outperforms SERPT except in scenarios where the job processing time variance is extremely low. Examining the SEH policy under other error models as well as analytic bounds as to how far it is from optimal could be investigated in future work. It would also be useful to examine how well policies designed for worst case performance would perform with respect to the performance metrics considered in this chapter. The work of Purohit et al. (27) is an intriguing candidate, as it runs two policies in parallel to provide worst case performance guarantees, even when there are large estimation errors.

Not much work has been done in the area of multi-server scheduling in the presence of estimation errors. One major reason is that determining optimal policies for multi-server queues is much more challenging compared to the single-server case. Mailach and Down (28) suggest that when SRPT is used in a multi-server system, the estimation error affects the system's performance to a lesser degree than in a single-server system. Grosof et al. (29) prove that multi-server SRPT is asymptotically optimal when an M/G/k system is heavily

loaded. Our work only evaluates the performance of SEH in a single-server framework so we leave the extension and evaluation of this policy in multi-server queues for future investigation.

Acknowledgment

The authors would like to thank Ziv Scully for useful discussions on the limitations of the SEH policy.

Bibliography

1. M. Akbari-Moghaddam and D. G. Down, "SEH: Size Estimate Hedging for Single-Server Queues," in *International Conference on Quantitative Evaluation of Systems*. Springer, 2021, pp. 168–185.

2. L. Schrage, "Letter to the editor—a proof of the optimality of the shortest remaining processing time discipline," *Operations Research*, vol. 16, no. 3, pp. 687–690, 1968.

3. L. E. Schrage and L. W. Miller, "The queue M/G/1 with the shortest remaining processing time discipline," *Operations Research*, vol. 14, no. 4, pp. 670–684, 1966.

4. M. Mitzenmacher, "Scheduling with predictions and the price of misprediction," *arXiv preprint arXiv:1902.00732*, 2019.

5. D. Lu, H. Sheng, and P. Dinda, "Size-based scheduling policies with inaccurate scheduling information," in *The IEEE Computer Society's 12th Annual International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems, 2004.(MASCOTS 2004). Proceedings.* IEEE, 2004, pp. 31–38.

6. R. Mailach, "Robustness to estimation errors for size-aware scheduling," Ph.D. dissertation, McMaster University, Department of Computing and Software, Canada, 2017.

7. M. Dell'Amico, "Scheduling with inexact job sizes: The merits of shortest processing time first," *arXiv preprint arXiv:1907.04824*, 2019.

8. J. C. Gittins, "Bandit processes and dynamic allocation indices," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 41, no. 2, pp. 148–164, 1979.

9. S. Aalto, U. Ayesta, and R. Righter, "On the Gittins index in the M/G/1 queue," *Queueing Systems*, vol. 63, no. 1-4, p. 437, 2009.

10. M. Dell'Amico, D. Carra, and P. Michiardi, "PSBS: Practical size-based scheduling," *IEEE Transactions on Computers*, vol. 65, no. 7, pp. 2199–2212, 2015.

11. M. Harchol-Balter, B. Schroeder, N. Bansal, and M. Agrawal, "Size-based scheduling to improve web performance," *ACM Transactions on Computer Systems (TOCS)*, vol. 21, no. 2, pp. 207–233, 2003.

12. H. Chang, M. Kodialam, R. R. Kompella, T. Lakshman, M. Lee, and S. Mukherjee, "Scheduling in mapreduce-like systems for fast completion time," in *2011 Proceedings IEEE INFOCOM*. IEEE, 2011, pp. 3074–3082.

13. A. Wierman and M. Nuyens, "Scheduling despite inexact job-size information," in *Proceedings of the 2008 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*, 2008, pp. 25–36.

14. M. A. Bender, S. Muthukrishnan, and R. Rajaraman, "Improved algorithms for

stretch scheduling," in *Proceedings of the thirteenth annual ACM-SIAM symposium on Discrete algorithms*, 2002, pp. 762–771.

15. L. Becchetti, S. Leonardi, A. Marchetti-Spaccamela, and K. Pruhs, "Semi-clairvoyant scheduling," *Theoretical computer science*, vol. 324, no. 2-3, pp. 325–335, 2004.

16. M. Pastorelli, A. Barbuzzi, D. Carra, M. Dell'Amico, and P. Michiardi, "HFSP: size-based scheduling for Hadoop," in *2013 IEEE International Conference on Big Data*. IEEE, 2013, pp. 51–59.

17. A. Wierman, "Fairness and scheduling in single server queues," *Surveys in Operations Research and Management Science*, vol. 16, no. 1, pp. 39–48, 2011.

18. Z. Scully, M. Harchol-Balter, and A. Scheller-Wolf, "Soap: One clean analysis of all age-based scheduling policies," *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, vol. 2, no. 1, pp. 1–30, 2018.

19. M. E. Crovella, M. S. Taqqu, and A. Bestavros, "Heavy-tailed probability distributions in the World Wide Web," *A practical guide to heavy tails*, vol. 1, pp. 3–26, 1998.

20. M. Harchol-Balter, "The effect of heavy-tailed job size distributions on computer system design." in *Proc. of ASA-IMS Conf. on Applications of Heavy Tailed Distributions in Economics, Engineering and Statistics*, 1999.

21. Y. Chen, S. Alspaugh, and R. Katz, "Interactive analytical processing in big data systems: A cross-industry study of mapreduce workloads," *arXiv preprint arXiv:1208.4174*, 2012.

22. I. A. Rai, G. Urvoy-Keller, and E. W. Biersack, "Analysis of LAS scheduling for job size distributions with high variance," in *Proceedings of the 2003 ACM SIGMETRICS*

international conference on Measurement and modeling of computer systems, 2003, pp. 218–228.

23. E. G. Coffman and P. J. Denning, *Operating systems theory*. prentice-Hall Englewood Cliffs, NJ, 1973, vol. 973.

24. L. Kleinrock, "Queueing Systems: vol. 1, Theory," 1975.

25. R. Righter and J. G. Shanthikumar, "Scheduling multiclass single server queueing systems to stochastically maximize the number of successful departures," *Probability in the Engineering and Informational Sciences*, vol. 3, no. 3, pp. 323–333, 1989.

26. S. Yashkov, "Processor-sharing queues: Some progress in analysis," *Queueing Systems*, vol. 2, no. 1, pp. 1–17, 1987.

27. M. Purohit, Z. Svitkina, and R. Kumar, "Improving online algorithms via ML predictions," in *Advances in Neural Information Processing Systems*, 2018, pp. 9661–9670.

28. R. Mailach and D. G. Down, "Scheduling jobs with estimation errors for multi-server systems," in *2017 29th International Teletraffic Congress (ITC 29)*, vol. 1. IEEE, 2017, pp. 10–18.

29. I. Grosof, Z. Scully, and M. Harchol-Balter, "SRPT for multiserver systems," *Performance Evaluation*, vol. 127, pp. 154–175, 2018.

Chapter 3

Data-driven Fair Resource Allocation For Novel Emerging Epidemics: A COVID-19 Convalescent Plasma Case Study

This chapter is adapted from Maryam Akbari-Moghaddam1, Na Li, Douglas G. Down, Donald M. Arnold, Jeannie Callum, Philippe Bégin, and Nancy M. Heddle, "Data-driven Fair Resource Allocation For Novel Emerging Epidemics: A COVID-19 Convalescent Plasma Case Study", submitted to Health Care Management Science, and available on arXiv, arXiv:2106.14667v1. Additional discussions on resource forecasting are included throughout the chapter.

Abstract

Epidemics are a serious public health threat, and the resources for mitigating their effects

are typically limited. Decision-makers face challenges in forecasting the demand for these resources as prior information about the disease is often not available, the behaviour of the disease can periodically change (either naturally or as a result of public health policies) and can differ by geographical region. In this work, we discuss a model that is suitable for short-term real-time supply and demand forecasting during emerging outbreaks without having to rely on demographic information. We propose a data-driven mixed-integer programming (MIP) resource allocation model that assigns available resources to maximize a notion of fairness among the resource-demanding entities. Numerical results from applying our MIP model to a COVID-19 Convalescent Plasma (CCP) case study suggest that our approach can help balance the supply and demand of limited products such as CCP and minimize the unmet demand ratios of the demand entities.

Keywords: Resource Allocation, Epidemics, COVID-19 Convalescent Plasma, Data-driven Optimization, Demand Forecasting.

3.1 Introduction

Epidemics have impacted the world many times, and will occur again in the future. Emergency responses to these epidemics can either be pre-event or post-event. Forecasting the potential dangers and planning the necessary steps to deal with an epidemic are considered as pre-event tasks. Post-event responses occur after the disease starts spreading and is still in progress. The corresponding actions at these points are associated with treatment and allocating the corresponding available resources. Our focus in this chapter is on post-event response situations.

Resource allocation decisions during emerging epidemics are challenging due to several key reasons. First, limited knowledge and historical data about disease demographics make it difficult to predict the demand for particular resources. Second, since epidemics are usually unexpected and can spread rapidly, there are often limited health care resources (vaccines, blood products, medical equipment, etc.) compared to the total number of entities requesting them. Determining how to fairly allocate the limited resources becomes a challenge. Third, the demand can vary significantly between geographically dispersed entities. Finally, the decisions must be made in a timely manner. These issues motivate the investigation of supply and demand forecasting models for scenarios where there are small amounts of available data and the data can exhibit fundamental changes in behaviour. It is of interest to incorporate these models into algorithms that yield fair allocations.

In this work, we tackle the real-time allocation of scarce resources during epidemics to entities located in widespread geographical locations and with different resource requirements. We are interested in demand forecasting models that can predict short-term demand in real-time. The demand forecasts directly impact the resource allocation decisions as inaccurate forecasts may lead to inefficient and unfair use of limited (and valuable) resources. We note that there are different notions of fairness (balance) in terms of resource allocation in the literature (1). In our case, we define fairness as minimization of the entities' unmet demand ratios, but one could also generalize fairness to other notions. For instance, Karsu et al. (2) propose an approach where imbalance is defined as the deviation from a reference distribution determined by the decision-maker. They show that in resource allocation problems, it is possible to maintain a mixed-integer programming (MIP) structure even after generalizing the notion of fairness. In general, there are wide-ranging views of fairness and how fairness metrics can be incorporated into optimization problems, see (3; 4; 5; 6; 7) for

example.

Convalescent plasma has been used as a potential treatment for a number of diseases such as Ebola (8; 9), influenza (10; 11), and COVID-19 (12). COVID-19 Convalescent Plasma (CCP), also known as "survivor's plasma," contains antibodies, or special proteins, generated by the body's immune system in response to the novel coronavirus. It has been considered as an experimental treatment for hospitalized COVID-19 patients in a number of randomized control trials worldwide. We evaluate our proposed model on a case study of CCP distribution within a clinical trial when there were limited historical supply and demand data, the supply was limited and restricted by manufacturing policies, the demand arising from the demand entities was heterogeneous, and specific clinical requirements were needed for administrating CCP transfusion.

We make the following contributions: First, we discuss real-time short-term forecasting of supply and demand of scarce resources in epidemics with sparse data, no historical data and without relying on epidemiological models or demographic information. We propose the use of a forecasting model that does not require indeterminate parameters (such as location and time-specific parameters) and thus does not require periodically updating the parameters. Secondly, we address challenges that may arise in an online setting due to extrapolation and sparse data. Next, we propose a data-driven MIP model for real-time multi-location allocation of scarce resources regularly and fairly to entities, which have heterogeneous demand. This approach maximizes a notion of fairness among the resource-demanding entities. Finally, numerical results of applying our model in a CCP case study show that our approach yields fair allocations that are both close to the scenario where supply and demand are known (rather than forecast) and are preferable to what was used in practice.

The rest of the chapter is organized as follows. Section 3.2 presents the existing

literature on demand forecasting and resource allocation approaches during infectious disease outbreaks and our motivation for this work. We describe our data-driven resource allocation problem in Section 3.3.1. Section 3.3.2 discusses in depth the supply and demand forecasting methods that we use and we define our proposed MIP resource allocation model in Section 3.3.3. The CCP case study and the numerical results of applying our MIP model to the case study are discussed in Section 3.4. We conclude this work and discuss how it may inform responses to future pandemics in Section 3.5.

3.2 Motivation and Related Work

Epidemiological compartmental models, consisting of a set of nonlinear ordinary differential equations, can help model the dynamics of different epidemiological variables during a pandemic (13). These models can give insight into disease-related information such as spread rate, the duration of an epidemic, and the total number of infected and recovered patients. Decision-makers can employ compartmental models to derive demand for medical resources to guide resource allocation decisions. Focusing on the COVID-19 outbreak (14), there have been many applications and tools developed by different organizations worldwide to forecast infections, hospitalizations, and deaths using compartmental models (15; 16; 17). For instance, CHIME (18) is a tool based on a Susceptible-Infectious-Recovered (SIR) model that can be used for forecasting the number of daily hospitalized COVID-19 patients in the short-term (e.g., up to 30 days). When an epidemic is first emerging, the epidemic state is only partially observable, and the parameters that the epidemiological models require are often indeterminate, as the disease information can only be obtained over a period of time and after sufficient cases are reported and required data is collected. Moreover, different geographical locations can show various characteristics in terms of the disease

spread pattern. The estimates that compartmental models make are sensitive to the model's structure (19). Even in the presence of reasonable parameter estimates and simple model structures, real-time resource allocation requires the parameters to be periodically updated based on the disease spread rate and the number of people involved, whether susceptible or infected. Therefore, CHIME-like models may lead to poor approximation of the actual demand when used in a real-time setting where obtaining the most recent updated parameters may not always be possible.

Another approach that researchers have studied for forecasting healthcare resources is using time series models or machine learning methods. The references for this approach are extensive, thus we only discuss a few studies as examples. Ferstad et al. (20) introduce a time series model to forecast the availability and utilization of intensive and acute care beds. Nikolopoulos et al. (21) use epidemiological and deep learning models to forecast the excess demand for products and services considering auxiliary data and simulating governmental decisions, while Li et al. (22) combine ideas from statistical time series modelling and machine learning to develop a hybrid demand forecasting model for red blood cell components using clinical predictors. All of these methods work best when large datasets are available and the models can capture the trend and seasonality, which is not possible during emerging epidemics. Furthermore, real-time demand forecasting can be challenging with any forecasting model if the demand is affected by many external factors, such as population characteristics, geographical locations, operational procedures, guidelines, and governmental policies.

As far as we are aware, none of the mentioned approaches are consistent with the challenges we introduced in Section 3.1. It is quite difficult to directly apply these approaches to a real-time setting where short-term supply and demand forecasting is desired, there is a

limited supply for resources, and the available data is sparse and shows fundamental changes in behaviour during different periods. This motivates us to avoid models that are reliant on a large number of parameters (as for CHIME-like models), as they require continual updates and they may not always yield accurate forecasts for resource supply and demand. Nonetheless, we seek a model that makes reasonable predictions even when facing such challenges. We will revisit these challenges in the case study in Section 3.4.

We find piecewise linear regression (PLR), also known as segmented linear regression, a reasonable forecasting model for our settings. PLR forecasting models are a special case of a larger set of models known as spline functions (23). Modelling the regression function in "pieces" can be helpful when dealing with sparse data since we can still use linear regression models for data that does not fit a single line. To be more specific, PLR is a simple model that makes understanding the data easier by solving several linear regressions. Points at which the behaviour changes are called breakpoints, which act as boundaries between each piece. There have been a few studies on finding the number of breakpoints and their locations. Rosen and Pardalos (24) propose a method for finding the minimum number of equally spaced breakpoints within a given error tolerance, a sequential method is proposed in Strikholm (25) for finding the number of breakpoints, and Yang et al. (26) propose a discontinuous piecewise linear approximation and how to determine the optimal breakpoint locations.

Piecewise linear models have been used in different applications when modelling the structural shifts in data and forecasting based on the most recent behaviour in data is desired. Hong et al. (27) use a piecewise linear function for modelling the hourly demand for electric load and investigating the causality of the consumption of electric energy. In the domain of strategic product planning, Huang and Tzeng (28) propose a two-stage fuzzy piecewise

regression method to predict product life time and annual shipments of products during the product life cycle of multigeneration products. PLR has also been used in stock forecasting studies. For instance, Chang et al. (29) apply PLR to historical stock data to decompose them into different segments and detect the temporary (trough or peak) turning points. They give these points as inputs to a backpropagation neural network model to train a pattern matching model for the stock market. We will discuss two PLR models, MARS and PLR-NB in more detail in Section 3.3.2 and Section 3.3.2, respectively, where we discuss supply and demand forecasting models used in this work. The MARS model has been used in healthcare for medical diagnosis using classification problems (see (30; 31; 32; 33; 34; 35)). In the area of time series forecasting, López-Lozano et al. (36) evaluate the generalizability of MARS for identifying thresholds for antibiotic consumption and Katris (37) studies MARS and other time series approaches for predicting the evolution of reported COVID-19 cases to track the outbreak in Greece. In this work, we demonstrate that using PLR models to determine inputs (supply and demand forecasts) to a resource allocation problem is an effective combination in an emerging epidemic setting.

Many studies have focused on developing allocation models for medical resources during infectious disease outbreaks. A dynamic linear programming model based on an epidemic diffusion model is introduced in Liu et al. (38) to allocate medical resources. Preciado et al. (39) analyze a networked version of a susceptible-infected-susceptible (SIS) epidemic model when different susceptibility levels are present. They propose a convex optimization approach for distributing vaccination resources in a cost-optimal manner and test their approach in a real social network. Yarmand et al. (40) consider two-phase vaccine allocation to different geographical locations. They capture each region's epidemic dynamics for different vaccination phases by a two-stage stochastic linear program (2-SLP) model

and show that their model helps to reduce vaccine production and administration costs. Furthermore, two resource allocation problems during outbreaks are discussed in Preciado et al. (41), where they use geometric programming to solve the problems. Following the work in (41), Han et al. (42) propose a data-driven robust optimization framework based on conic geometric programming. Their model is used to determine an optimal allocation of medical resources such as vaccines and antidotes and can help control an SIS viral spreading process in a directed contact network with unknown contact rates.

A number of works have investigated resource allocation frameworks using outbreak case studies. The problem of scheduling limited available resources between multiple infected areas is discussed in Rachaniotis et al. (43), and their proposed deterministic scheduling model is studied in a case study of mass vaccination against A(H1N1)v influenza. A real-time synchronous heuristic algorithm is proposed in (44) and is tested on the same case study as in (43). Sun et al. (45) focus on allocating patients and resources between hospitals located in a healthcare network and propose a multi-objective optimization model. They discuss the application of their model in an influenza outbreak case study. Finally, a large integer programming problem framework for optimally allocating a resource donation is introduced in Anparasan et al. (46), and results of applying the framework to a 2010 cholera outbreak case study are reported. Closely related to our work is Du et al. (47) where they study a multi-period location-specific resource allocation problem for cholera outbreak intervention. They consider a rolling time horizon and periodically determine an optimal intervention resource allocation strategy with their data-driven optimization approach. Also similar in spirit to our approach is Bekker et al. (48), who propose a model for making daily short-term predictions of the number of occupied ICU and clinical beds in the Netherlands due to COVID-19. Their prediction model consists of a linear programming model inspired

by smoothing splines for predicting the arrivals and methods stemming from queueing theory to convert arrivals into occupancy. The motivation for choosing their model is similar to ours in the sense that it works with little historical data, which is a consequence of an emerging epidemic setting. To the best of our knowledge, no other study has tackled the problem of real-time multi-location allocation of scarce resources with sparse historical data and without relying on epidemiological models.

3.3 Data-driven Resource Allocation Model

3.3.1 Problem Description

We are interested in a setting where limited resources must be allocated on a regular basis (e.g. every week) to the entities requesting them, the demand for the resources can be heterogeneous and arises from geographically dispersed locations. We consider a hub-and-spoke structure where we have a centralized supplier (hub) that interacts with H customers (spokes) and is responsible for satisfying their demand for R types of resources. Figure 3.1 shows a flowchart of our data-driven resource allocation process.

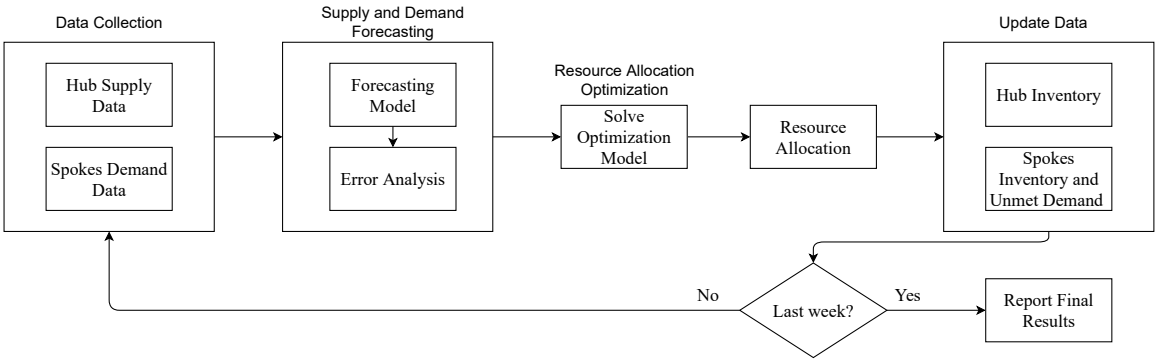


Figure 3.1: Data-driven Resource Allocation Process

We choose to work with the cumulative supply or demand to deal with the data sparsity

effect. Working with a cumulative sum of data samples over time is helpful in situations where one needs to smooth heterogeneous and sparse data and still make quantitative predictions for future supply and demand without altering the original data. Ellaway (49) investigates the application of the cumulative sum technique in a neurophysiology study. The cumulative sum is shown to be a powerful technique for finding periods of change in data, as well as reducing real-time decision-making uncertainty. In our case, we forecast weekly supply and demand based on historical cumulative supply and demand data for a particular resource prior to resource allocations. In particular, we would like to fit a forecasting model weekly to our data where for each week, only the last week's observation is added to the dataset, and we cannot modify the predictions the model previously made (as real-time allocations are made based on the forecasts). We then perform error analysis of the forecasts, and allocate the available resources to the customers in a fair manner by solving an appropriate optimization problem. We assume that both the supplier and customers can hold inventories of the resources and can use them to satisfy future demand. Once the supplier allocates resources to customers, the decision is final, and the resources cannot be reassigned. Thus, we need to update the inventories at the end of each week, and consider them when solving the optimization model in the next week.

We discuss two PLR supply and demand forecasting models and our resource allocation optimization model in more detail in Section 3.3.2 and Section 3.3.3, respectively.

3.3.2 Supply and Demand Forecasting

We would like to forecast the supply and demand for a particular week $t+1$ where we only have data available up to week t. Consider a dataset consisting of t observations where X is an array of elements x_i ($i = 1,2,\ldots,t$) representing the input variables (in our case,

$X = 1,2,3,\ldots,t$). Consider y to be a vector of observed (supply or demand) data samples y_i ($i = 1,\ldots,t$). We use two PLR models, Multivariate Adaptive Regression Splines (MARS) and a simplified PLR (PLR-NB) to forecast the supply and demand.

Multivariate Adaptive Regression Splines (MARS)

Multivariate Adaptive Regression Spline (MARS) is a nonparametric regression approach that was introduced by Friedman (50). The MARS model consists of a collection of simple linear models that can capture patterns and trends related to interactions and nonlinearities. MARS uses a series of piecewise linear pieces (splines) of different gradients. These pieces, also known as basis functions (BFs), are connected at positions called knots which allow thresholds, bends, and other departures from linear functions. A MARS model is specified as follows:

$$\hat{y}' = \beta_0 + \sum_{p=1}^{P} \beta_p \lambda_p(x),$$

where P is the number of BFs and each $\lambda_p(x)$ is a BF, which can be a spline function or the product of two or more spline functions. The parameters β_0 and β_p, $p = 1,\ldots,P$ are estimated using the least-squares method. The basis functions are described as:

$$BF(x) = \{\max(0, x - c_p), \max(0, c_p - x)\},$$

in which, c_p is the knot of the spline (threshold value).

A forward stage and a backward stage are considered in the MARS algorithm. In the forward stage, BF functions and their potential knots are chosen, which may result in a complicated and over parameterized model. In the backward stage, to prevent overfitting,

the model considers deleting the BFs in increasing order of the amount that they reduce the training error (50).

MARS determines the optimal number and locations of the knots (here, knots are considered as breakpoints), which can introduce computational overhead. One simplification to a PLR model is to provide the number of breakpoints to the model. We use the method proposed in Golovchenko (51) to find the breakpoint locations given that we know the number of breakpoints. We refer to this approach as PLR-NB and discuss it in the following.

PLR-NB

PLR-NB is a simplified PLR model as the number of breakpoints is an input for the model. PLR-NB can be a suitable choice for situations where sufficient information about the time series characteristics (such as trend and seasonality) is available. Since the number of breakpoints can be determined by observing changes in the behaviour of the time-series data, PLR-NB only searches for the best location of the breakpoints, skipping the computational overhead that MARS has for determining this number. We now discuss this method in more detail.

In the PLR-NB approach, α breakpoints are enforced on X. Every possible combination should be checked in order to find the best breakpoint locations. Denote by B a choice of breakpoints $\{b_1, \ldots, b_\alpha\}$. We also define $b_0 = 0$ and $b_{\alpha+1} = t$ (independent of B).

A simple linear regression is performed for each piece, and the total error is calculated as:

$$\delta_B = \sum_{k=1}^{\alpha+1} \sum_{j=b_{k-1}+1}^{b_k} (f_k(x_j) - y_j)^2$$

where $f_k(x_j)$ is the predicted value for x_j with respect to the linear regression fitted to its

corresponding piece k.

We note again that this procedure is done for all possible sets of breakpoints, B. The combination that produces the minimum δ_B will specify the breakpoint locations. Finally, the breakpoint indexes associated with the minimum δ_B are used to calculate the supply or demand forecasts $\hat{y}'_i = f_k(x_i)$ for each x_i in X based on its corresponding piece k.

One drawback of PLR-NB is that the model needs to know α beforehand. However, it can be a useful tool for evaluating the data based on a specific number of different pieces while maintaining the model's interpretability, for example, when the implementation dates of new procedures or policies are known.

Error Analysis and Model Enhancement

Based on the slope of the last piece, the forecast supply and demand for week $t+1$ under both MARS and PLR-NB is simply:

$$\hat{y}'_{t+1} = \hat{y}'_t + \frac{\hat{y}'_t - \hat{y}'_{t-1}}{x_t - x_{t-1}}.$$

We can improve the forecast value for week $t+1$ ($\hat{y}'_{t+1}$) by calculating its forecast error (52; 53). We first fit an autoregressive (AR) model of order l using Conditional Maximum Likelihood to the residual error ($\varepsilon_i = y_i - \hat{y}'_i$) data up to week t and use it to forecast the error for week $t+1$:

$$\hat{\varepsilon}_{t+1} = a_0 + a_1\varepsilon_t + a_2\varepsilon_{t-1} + \cdots + a_l\varepsilon_{t-l}, \tag{3.3.1}$$

where $a_0, a_1, \ldots, a_l$ are the coefficients obtained from the AR model and $\hat{\varepsilon}_{t+1}$ is the forecast error for week $t+1$. We finally calculate $\hat{y}_{t+1} = \hat{y}'_{t+1} + \hat{\varepsilon}_{t+1}$, i.e., the improved supply or

demand forecast value for week $t+1$, and use it as our final forecast value for that week.

Challenges

We now discuss general challenges that may be faced when MARS and PLR-NB are employed in an online manner:

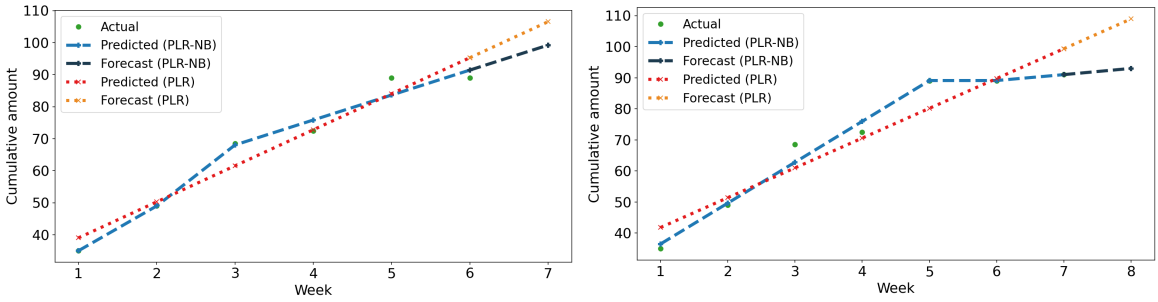
- Choosing between MARS and PLR-NB mostly depends on the setting of interest. In particular, PLR-NB may be a better choice when seasonality in the data can be captured, examining the data under a specific number of breakpoints is desired, for example as a result of underlying knowledge about the data. However, MARS is more suitable for real-time settings and uncertain situations where insufficient information is available to identify the number of breakpoints.

- Predictions made using MARS and PLR-NB may degrade if there is a sudden transitory change in data. For instance, we found that holidays may affect the amount of data collected in a particular week but the data follows its previous pattern after the holidays have passed. We discuss this issue in greater detail in the case study discussed in Section 3.4.

- Both MARS and PLR-NB are fitting piecewise linear models, and thus the slope of the segment that ends with the most recent observation has a significant effect on the forecast for the next week (as it may lead to a large underestimation/overestimation).

Although working with the cumulative sums for forecasting future supply and demand can help address the issue of data sparsity, a particular challenge arises when cumulative sums are employed in an online setting. Consider a situation where the demand before week t has a steep slope resulting in a (relatively) high forecast for week t. However, upon

observing the demand for week t, one finds out that the actual demand was considerably lower. Thus, the slope that affects week $t+1$'s forecast demand will be less steep than what it was when only data up to week $t-1$ was available. This may result in the forecast cumulative demand for week $t+1$ being lower than the previously forecast cumulative demand for week t, which is not possible. For instance, Figure 3.2a and Figure 3.2b demonstrate the forecast value obtained by MARS and PLR-NB (with one breakpoint) for week 7 and week 8, respectively. The cumulative forecast value for week 7 and week 8 are 99 and 93, respectively, under PLR-NB (this issue can also arise under MARS), which cannot happen in practice. This is one consequence of considering an online setting where modifying the predictions is not possible as real-time decisions are made. The sparser the data, the more this issue can affect the models' predictions. A possible solution for dealing with such situations is to assume that the cumulative forecast value for week $t+1$ is equal to the cumulative forecast value for week t. We will observe this issue in our case study discussed in Section 3.4 and deal with it in the suggested manner.



(a) Forecast value for week 7

(b) Forecast value for week 8

Figure 3.2: Forecasting a negative non-cumulative value

3.3.3 Resource Allocation Optimization

We allow for demand for resource r to be satisfied by resource r'. This can be represented with a matrix C of size $R \times R$, where an element (r,r') of C is 1 if demand for resource r can be satisfied by resource r' and is 0 otherwise. Furthermore, the supply used in our MIP model for resource r for week t is constrained as follows:

$$c_{r,t} = \begin{cases} \hat{s}_{r,t}, & \text{if } \hat{s}_{r,t} \leq s_{r,t} \\ s_{r,t}, & \text{otherwise,} \end{cases} \tag{3.3.2}$$

where $s_{r,t}$ and $\hat{s}_{r,t}$ are the actual and forecast supply for resource r on week t, respectively. The second case in (3.3.2) indicates that we cannot allocate resources beyond the actual available supply. We will formulate our resource allocation as a Mixed Integer Program (MIP). To do so, we require the following notation:

Indices

t index of time periods, $t = 1,\ldots,T$

h index of customers, $h = 1,\ldots,H$

r, r' index of resource, $r, r' = 1,\ldots,R$

Data

$c_{r,t}$ the amount of resource r available at the supplier for assignment at time t (see (3.3.2) above)

$i_{r,h,t-1}$ the inventory of resource r stored at customer h at time $t-1$

$\hat{d}_{r,h,t}$ the estimated demand for resource r by customer h at time t

Decision variables

$v_{r,r',h,t}$ the number of units of resource r' assigned to customer h to satisfy demand for resource r at time t. We only consider the set of $v_{r,r',h,t}$ that correspond to $C(r,r') = 1$.

We formulate our objective function as follows:

Objective function

$$\min \max_{h:\sum_{r=1}^{R} \hat{d}_{r,h,t}>0} \frac{\sum_{r=1}^{R}(\hat{d}_{r,h,t} - \sum_{r'=1}^{R} v_{r,r',h,t} - i_{r,h,t-1})}{\sum_{r=1}^{R} \hat{d}_{r,h,t}}. \tag{3.3.3}$$

The objective function (3.3.3) captures our notion of fairness: minimizing the largest ratio of unmet demand over all customers. It could be modified to capture other notions of fairness.

Constraints

$$\sum_{h=1}^{H} \sum_{r=1}^{R} v_{r,r',h,t} \leq c_{r',t} \ , \forall r', \tag{3.3.4}$$

$$v_{r,r',h,t} \geq 0 \text{ and integer valued}, \tag{3.3.5}$$

$$\sum_{r'=1}^{R} v_{r,r',h,t} \leq \hat{d}_{r,h,t} \quad , \forall r \ , \forall h. \tag{3.3.6}$$

Constraint (3.3.4) prevents the over-allocation of available resources. Constraint (3.3.5) ensures the integrality and non-negativity of the resource allocations and constraint (3.3.6) keeps the resource allocated to each customer below the corresponding estimated demand. If all of the estimated demand at time t can be met, any excess supply is held in inventory at the hub.

3.4 The CONCOR-1 trial: A case study for a proposed application of the resource allocation model

COVID-19 Convalescent Plasma (CCP) has been assessed as an experimental treatment in a number of randomized control trials worldwide (54).

The Randomized, Open-Label Trial of CONvalescent Plasma for Hospitalized Adults With Acute COVID-19 Respiratory Illness (CONCOR-1) was a randomized clinical trial (RCT) involving 72 academic and community sites across Canada, the USA, and Brazil (55). The randomization in this RCT was performed at a ratio of 2:1 allocation to receive CCP or standard of care for a planned study population of 1200 patients, stratified by age (< 60 and ≥ 60 years). The first CCP unit for the trial was collected on April 24, 2020, and the first patient was randomized on May 14, 2020. The trial ceased on January 29, 2021 with a total of 940 randomized patients. The objective of the trial was to assess whether transfusing CCP reduces the proportion of patients requiring intubation or deaths at day 30 compared to standard of care for hospitalized COVID-19 infected adult patients (55). The CONCOR-1 team at the McMaster Centre for Transfusion Research and Canadian Blood Services were responsible for Canada's (excluding Québec) supply and demand management of the CCP products for patients enrolled in the trial. In what follows, as we are evaluating our approach, we will take the viewpoint that the trial is in progress.

Canadian Blood Services (CBS) is responsible for collecting the CCP units and is the national blood supplier across all provinces in Canada except Québec. Canada's blood supply chain network is currently centralized and comprises two levels: regional CBS distribution sites and hospital blood banks. Nine CBS blood distribution sites are currently located across Canada, and each centre attempts to meet the CCP demand from the hospital

blood banks in its network. We use all available data from the trial which comes from 18 hospital hubs, 30 hospital sites, and 8 CBS distribution sites.

CCP is stored frozen and ideally must be transfused as soon as possible after being thawed, but at most within five days (56). A patient who is randomized to the CCP arm in the trial requires a single dose of approximately 500 ml or two doses of 250 ml (from a single or two different donors) and the CCP unit is transfused to the patient within the first 24 hours after randomization (55).

It is challenging for CBS to make decisions on CCP allocation, since (i) the CCP supply is limited and restricted by manufacturing policies and may not meet the total CCP demand, (ii) the trial involves hospitals from geographically dispersed locations and multiple blood distribution centres, (iii) the demand for CCP exhibits heterogeneity between different geographic regions, (iv) there is limited knowledge or historical data about the disease demographics making it difficult to forecast the supply and demand of CCP products, (v) there are specific clinical requirements for administrating CCP transfusions, such as specific product dose, ABO blood group compatibility, and medical condition requirements, and (vi) the decisions must be made in real-time and once the CCP units are shipped to a hospital hub, redistribution to other hospital hubs is undesirable. Hence, the decision for every unit matters. The key observations of our work suggest that our proposed data-driven MIP model can help balance the supply and demand of CCP products and lead to a fair allocation of limited CCP products among hospital hubs.

The underlying network of our case study is shown in Figure 3.3. The corresponding definitions and notations are listed as follows:

- l: represents a cluster and is defined as a geographic region with only one CBS distribution site but one or more CBS donor collection sites, hospital hubs and individual

hospital sites.

- g: represents an individual CBS donor collection site in cluster l.
- b: represents the CBS distribution site in cluster l.
- h: represents an individual hospital hub under CBS distribution site b in cluster l.
- p: represents an individual hospital site for hospital hub h in cluster l.

Cluster *l*:

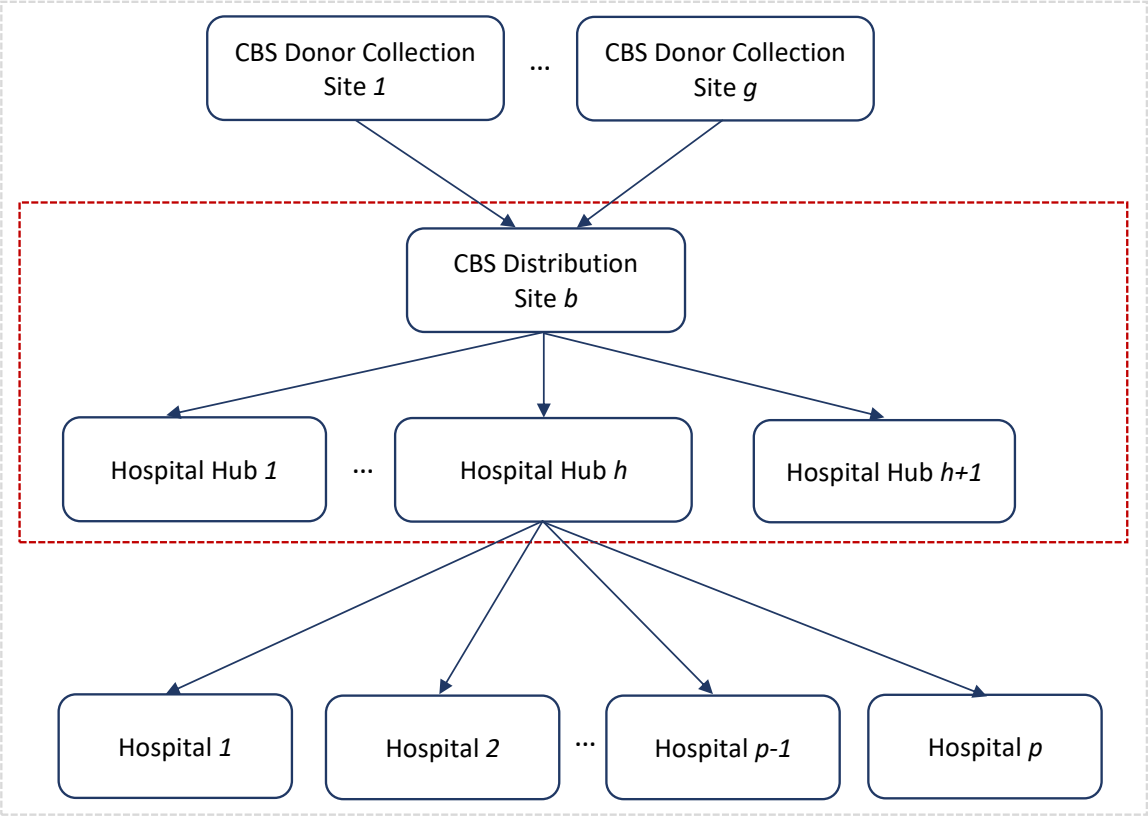


Figure 3.3: CCP allocation network

The structure in Figure 3.3 is a hub and spoke structure (a CBS distribution site is a hub and its underlying hospital hubs are the spokes) and is consistent with our resource

allocation MIP model proposed in Section 3.3.3. CBS distribution sites can centrally decide to reallocate the blood products if there is excess supply in a particular CBS region. Thus, we only consider a single CBS distribution site in our optimization problem. Based on the agreements between CBS and Héma-Québec, the blood supplier in Québec, less common blood groups (blood group AB and B) can be shared (55).

The CBS distribution site provides CCP to hospital hubs that in turn allocate units to other hospital sites in the area. We only focus on the allocations from the CBS distribution site to the hospital hubs.

As of May 2021, the distribution of ABO blood groups in Canada is O:46%, A:42%, B:9%, and AB:3% (57). In general, plasma for AB and O blood groups are universal donor and recipient, respectively. In practice, transfusing blood-specific units is prioritized. In the CONCOR-1 trial, due to the limited resources for B and AB plasma, patients with O and B blood groups can receive A and AB plasma, respectively, when an exact match is not available.

The dataset that we work with includes the CBS distribution site's available CCP units after shipment at aggregate level for each blood group on specific dates starting from September 1, 2020, up to January 25, 2021. It also contains data for the received CCP units from the CBS distribution site for each hospital hub, whether randomized to a patient or stored in inventory, from May 11, 2020, up to January 25, 2021. Furthermore, CCP-related information such as product dose, ABO group, and whether a unit was broken or leaking after being thawed for transfusion at a hospital hub are recorded. Table 3.1 provides a summary of the dataset's attributes, their description, and their format.

We first create a cumulative weekly dataset of the number of new 500 ml units assigned to randomized patients at each hospital hub (considered as their CCP demand) and the total

Dataset	Attribute	Description	Format
CBS supply data	date	CBS distribution site's aggregate available units after shipment on a date	Date
	A	Total number of blood group A CCP (250 ml) units	Integer
	AB	Total number of blood group AB CCP (250 ml) units	Integer
	B	Total number of blood group B CCP (250 ml) units	Integer
	O	Total number of blood group O CCP (250 ml) units	Integer
	Total	Total number of CCP (250 ml) units	Integer
Hospital Hub data	hospitalhub_ID	Unidentifiable unique ID of a hospital hub	String
	receiveddate	Hospital hubs received CCP units from the CBS distribution site on a date	Date
	DNL	The CCP unit's de-identified ID	String
	productABOgroup	The ABO blood group of the CCP unit	String
	productdose	The CCP unit dose with 1 indicating 250 ml units and 2 indicating 500 ml units	Integer
	matched	Boolean variable with 1 indicating the CCP unit was matched with a randomized patient (indicating a demand), 0 if the unit was stored in the hospital hub's inventory	Boolean
	thawed	Boolean variable with 1 indicating that the unit broke when thawing, 0 otherwise	Boolean

Table 3.1: Dataset description

received units from CBS for each resource. We cumulatively sum hospital hubs' weekly received CCP units and CBS weekly inventory difference to calculate CBS cumulative new weekly supply. The dataset is sparse in terms of the available information on CCP supply and demand. To maintain and promote efficient CCP allocation, prior estimation of the hospital hub's demand and CBS supply is necessary. One could either translate the CHIME model outputs to CCP supply or demand, or build a CHIME-like model that incorporates our variables. However, we found it more effective to work with the supply and demand directly and use PLR forecasting models. One reason is that the proportion of patients consenting to the trial also affects the total CCP demand. The consent rate is not random and can be affected by many factors such as a patient's religion and other competing treatment strategies. So, the patient population may not be the same as the population considered in the

CHIME or CHIME-like models. Furthermore, given the limited, sparse and highly-varied pattern of CCP supply and demand in our dataset, PLR models appear to be an effective tool for forecasting CCP units.

3.4.1 Model Assumptions

We have created an environment for weekly allocation of CCP units from the CBS distribution site to the hospital hubs participating in the CONCOR-1 trial. Our dataset contains CBS supply data from a later date than when the first patient was randomized to receive a CCP unit. So, we consider the week that CCP supply was first reported as the starting week in our model (week of August 31, 2020). The last week in the dataset is the week of January 25, 2021, so the duration T of our model is 22 weeks. We combined hospital hubs that are very close in distance, under the same distribution network and with few COVID-19 patients. Small hospitals in distant areas with a very low number of hospitalized COVID-19 patients were removed from the study dataset.

Table 3.2 shows the actual supply and demand (500 ml units) reported for each resource and in total in this period. There are a total of 15 A, 14 O, 9.5 B, and 14 AB 500 ml CCP units received from Héma-Québec recorded in the dataset. The numbers in Table 3.2 are only associated with CBS supply (excluding any units received from Héma-Québec). We observe in the data that 4.5 A, 3 O, 2 B, and 0.5 AB 500 ml CCP units are unused at the hospital hubs because they were leaking or broken after being thawed for transfusion. These units are not included in our model as the wastage due to the thawed CCP units is negligible.

CCP Units (500 ml)	A	O	B	AB	Total
Supply	157	84.5	29	26	296.5
Demand	131.5	94.5	34.5	33.5	294

Table 3.2: CBS CCP supply and CCP demand (August 31, 2020 - January 25, 2021)

We assume that we have no information on the actual CCP supply and demand when deciding on how to allocate the CCP units. Thus, we perform weekly forecasts considering only the available data up to that week. The open source *scikit-learn-contrib* library *py-earth* (58) is used for applying the MARS model, where the maximum degree of interaction terms generated by the forward stage (the *max_degree* parameter) is set to 2 to better deal with the nonlinearities in the data, and the remaining parameters follow the default values. We are not performing any validations since our dataset is sparse and the performance of the models cannot be significantly enhanced as using the cumulative sums has already smoothed the dataset. In general, a k-fold cross validation can be used with both MARS and PLR-NB to obtain a less biased estimate. The PLR-NB forecasting model, which requires specifying α, needs at least $\alpha+2$ data samples, so for ease of comparison with MARS, we start making predictions $(\alpha+2)+1$ weeks from the start of available data under both models. Based on the number of COVID-19 waves that occurred in the dataset period, we choose $\alpha = 1$ as the specified number of breakpoints in PLR-NB. Finally, a lag $l = 1$ is used in the autoregressive model of residual errors in (3.3.1). This choice is due to a week's supply and demand tending to be closer to the amounts for the most recent week, as well as trying to avoid sudden changes that we might face when considering a larger lag.

We deal with the challenges that might occur in our forecasting process, as discussed in Section 3.3.2; if a negative non-cumulative forecast value on week $t+1$ arises, we instead set the cumulative forecast value equal to the cumulative forecast value for week t. Furthermore, we account for expected sudden transitory changes in data, such as weeks containing holidays. We observe that the hospital hubs' requested CCP units are relatively lower on weeks containing the Christmas, Boxing Day, and New Year's Day statutory holidays. This issue is due to fewer working days or reduced staffing for these weeks. Thus,

it is important to account for these weeks as they can be falsely detected as breakpoints. We have chosen the value of 0.7 as a reasonable adjustment factor and multiply the slope of the forecast line going through the weeks containing Christmas, Boxing Day, and New Year's Day holidays by this factor. One might need different adjustment factors depending on the extent the supply and demand are affected on a particular holiday.

From this point on, wherever we refer to MARS and PLR-NB, we are considering the model's forecasts after accounting for the mentioned challenges and performing error analysis. Since CCP is stored in 250 ml units, we allocate 250 ml units from CBS to the hospital hubs in our model. The forecast values are decimal values; therefore, we round the supply and demand forecast values to the nearest integer so that all values correspond to 250 ml units. All the results reported in the figures and tables correspond to 500 ml units.

We are interested in making a fair allocation of different ABO blood group CCP units among the hospital hubs while minimizing their unmet CCP demand proportions. We consider four resources ($R = 4$) of A, O, B, and AB and examine two compatibility matrices C for assigning the CCP units in our MIP problem , as described in Section 3.3.3: (i) the identity compatibility matrix, and (ii) the ABO compatibility matrix used in the CONCOR-1 trial which allows the transfusion of A and AB plasma to patients with O and B blood groups, respectively, when the same blood group is not available (CONCOR-1 compatibility matrix). For our primary study, we use the forecast supply and demand for each of the resources in our model for each week. Furthermore, to evaluate different allocation scenarios, we also forecast the aggregate supply and demand for each week and consider the distribution of ABO blood groups in Canada as the probabilities for calculating the forecast CCP for each resource (59). For this sensitivity analysis, we use the following probabilities based on the distribution of Canadian blood groups on each run of our model for generating the supply and

demand for each blood group: $w_{c_A} = w_{h_A} = 0.42$, $w_{c_O} = w_{h_O} = 0.46$, $w_{c_B} = w_{h_B} = 0.09$, and $w_{c_{AB}} = w_{h_{AB}} = 0.03$, where w_{c_r} and w_{h_r} are the supply and demand probabilities for blood group r, respectively. The goal of the sensitivity analysis is to determine the performance of the model in situations where it is only possible to forecast the aggregate amount of resources in terms of supply and demand. This analysis also helps gain insights as to the generalizability of the results of our primary study.

We assume that both CBS and hospital hubs can store the excess units at the end of each week to use them in later weeks. We solve our MIP model based on forecast supply and demand for each resource, and the actual inventories held at CBS and hospital hubs on the week under study. We assume that the hospital hubs can use a compatible CCP unit according to the CONCOR-1 compatibility matrix (if available) for a patient when the same blood group CCP is not available. We examine both identity and CONCOR-1 compatibility matrices for solving our MIP model; however, at the inventory level of the hospital hubs, only the CONCOR-1 compatibility matrix is used. We shall see later that this is a good combination for the situations where forecast supply and demand are used, and the forecasting errors are not too large. The identity compatibility matrix at the MIP level prevents issues in terms of greedy allocation of scarce resources, and the CONCOR-1 compatibility matrix at the inventory level of the hospital hubs allows for the efficient compensation of forecasting errors.

Since we know the actual demand for a particular week only after that week has passed and since the supply is limited, we might not fully meet all demands. In these cases, the unmet demand is carried over to the next week. We note that the optimization model ensures that the CCP units of a resource allocated to a hospital hub are never more than its forecast demand. Excess units stored in hospital hubs' inventories are due to the possible difference

between the actual and forecast values.

A total of m runs are used to calculate the mean final unmet demand ($\overline{u}_T$) and the mean ratio of final unmet demand to total demand ($\overline{z}_T$) for each hospital hub:

$$\overline{u}_T = \frac{\sum_{r=1}^{R} u_{T_r}}{m}$$

where the duration of our model is $T = 22$ weeks, and u_{T_r} is the hospital hub's final unmet demand for resource r and,

$$\overline{z}_T = \frac{\sum_{r=1}^{R} u_{T_r}}{m \sum_{t=1}^{T} \sum_{r=1}^{R} d_{t_r}}$$

where d_{t_r} is the hospital hub's actual newly-added demand on week t for resource r.

3.4.2 Results

In Figure 3.4 and Figure 3.5, we show the (real-time) cumulative weekly supply and demand forecasts over time, respectively, in terms of total, A, O, B, and AB CCP units after using MARS and PLR-NB models for the dataset considering only the available data up to that week. The real-time demand forecasting for each week in Figure 3.5 uses the cumulative aggregate demand data over all hospital hubs up to that week. We observe that both models perform well even with our limited dataset. Table 3.3 and Table 3.4 show RMSE and MAPE of our supply and demand forecasts, respectively, under MARS and PLR-NB forecasting models for total, A, O, B, and AB CCP units; lower values are better. We observe that MARS fits the data well after performing error analysis and accounting for the mentioned challenges discussed in Section 3.3.2 and Section 3.3.2. We note that we are making forecasts from week 4, which explains the severe overestimation for week 4 in Figure 3.4e where the data is sparse. In such situations, the cumulative value remains the same until a subsequent

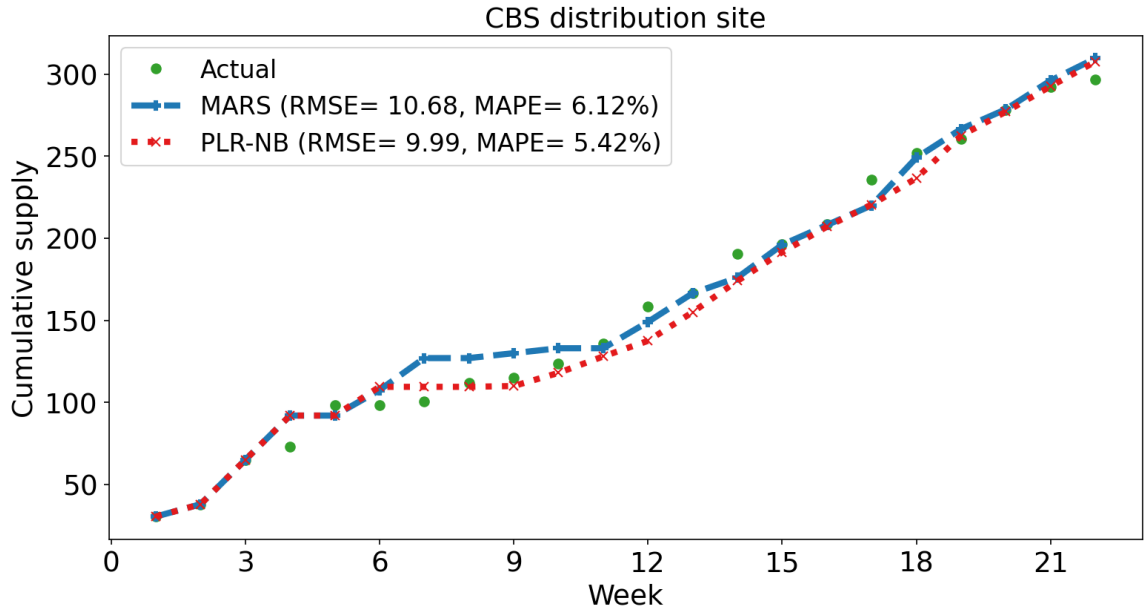
cumulative forecast is higher.

The models have similar performance in terms of their overall prediction errors. We observe that if there is only one sudden change in the behaviour of the data, enforcing only one breakpoint (as in PLR-NB) yields better results than MARS in some forecasts, but the difference is not significant. However, if more breakpoints are needed for the model to best fit the data, enforcing a specified number of breakpoints can result in significant errors. In such cases, MARS makes better forecasts by determining the optimal number of breakpoints. Therefore, we believe MARS is a better choice for this case study where the allocations are done in a real-time manner and the number of breakpoints can only be reasonably determined in hindsight.

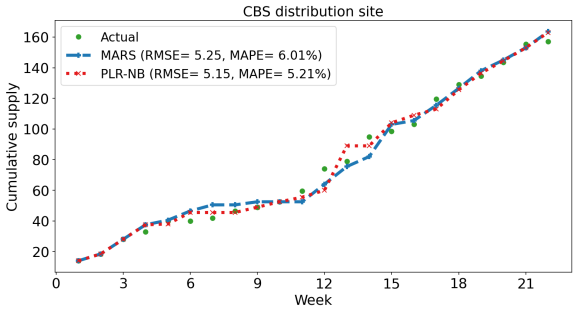
Hub	MARS					PLR-NB				
	Total	A	O	B	AB	Total	A	O	B	AB
Hospital Hub 1	1.06	0.43	0.84	0.43	0.41	1.51	0.53	0.95	0.52	0.60
Hospital Hub 2	1.74	1.21	0.91	0.84	0.58	1.36	0.87	0.65	0.72	0.81
Hospital Hub 3	2.97	1.88	1.56	0.81	0.87	2.98	1.47	1.77	0.69	0.83
Hospital Hub 4	2.33	1.59	0.95	0.61	0.49	2.12	1.06	1.43	0.72	0.85
Hospital Hub 5	1.76	1.10	0.80	1.02	0.46	1.72	1.48	0.82	1.21	0.43
Hospital Hub 6	3.00	1.38	1.73	1.06	0.66	3.26	1.45	1.48	1.12	0.83
Hospital Hub 7	2.64	2.01	1.04	-	-	1.95	2.37	0.79	-	-
All Hospital Hubs	7.01	4.27	1.99	1.57	1.57	7.33	4.06	1.76	1.30	1.71
CBS Distribution Site	10.68	5.25	5.41	1.78	3.35	9.99	5.15	6.43	2.36	3.33

Table 3.3: RMSE of supply and demand forecasts

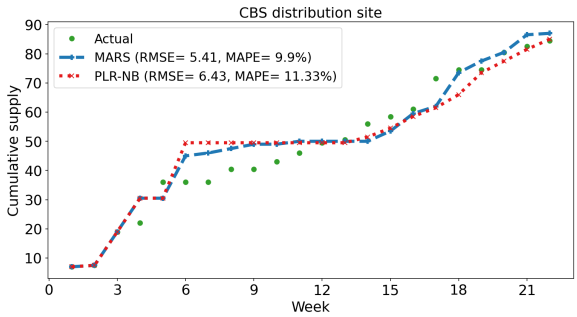
Table 3.6 and Table 3.7 report our CCP allocation model's performance under MARS and PLR-NB, respectively, in terms of $\overline{u}_T$, $\overline{z}_T$ for our primary study, i.e., allocating the resources based on the forecast supply and demand for each blood group. In Table 3.8, we compare the results to both when no forecasting is required, i.e., the actual values of supply and demand for each resource are known for each week and the actual allocations in the CONCOR-1 trial. In all of these three tables, two different compatibility matrices (identity and CONCOR-1) are chosen for the MIP allocation model. Finally, in Table 3.9 and Table



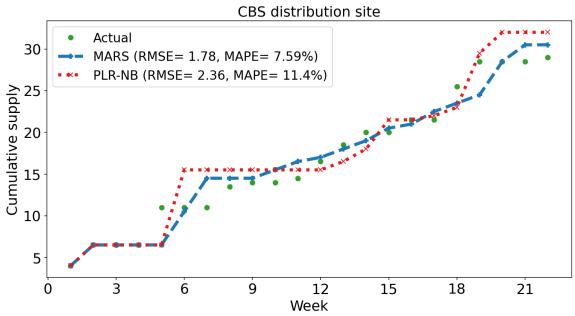
(a) Total units



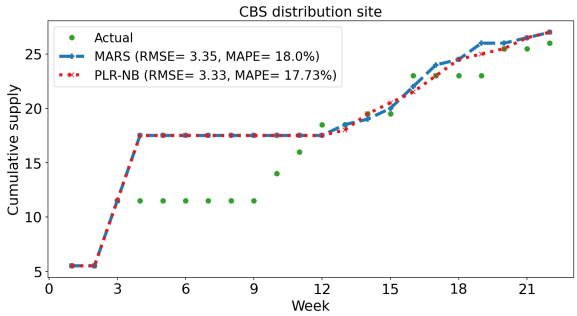
(b) A units



(c) O units

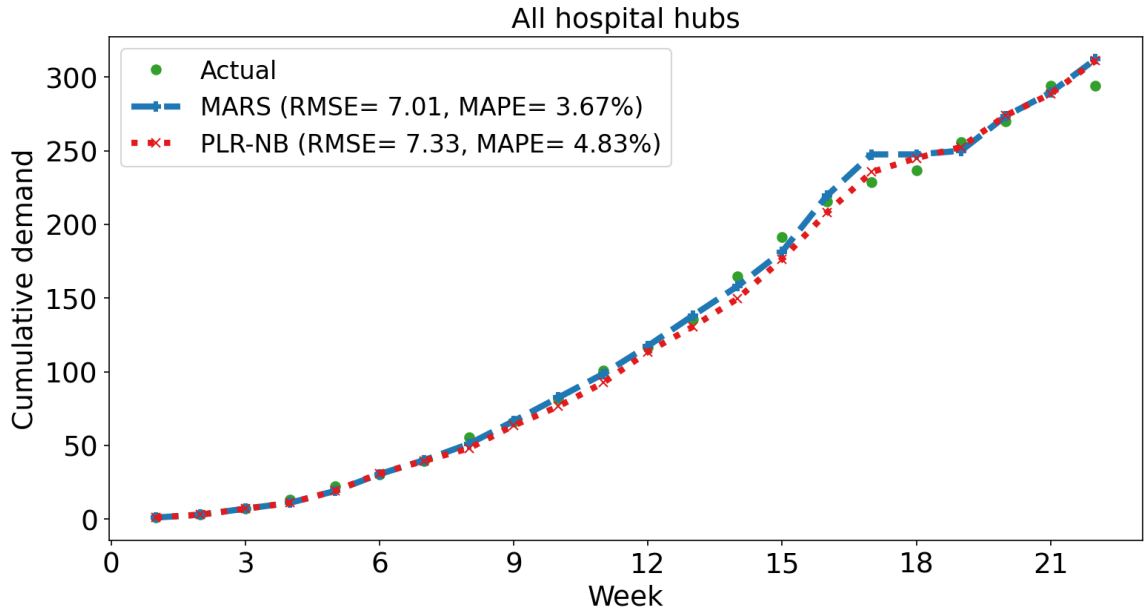


(d) B units

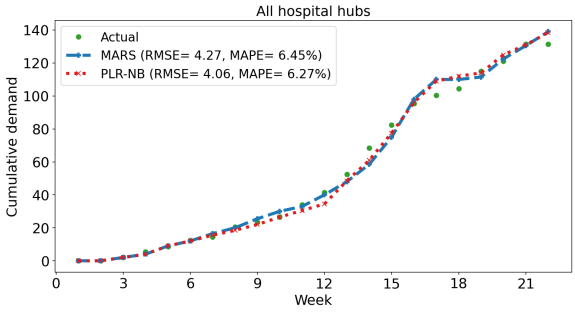


(e) AB units

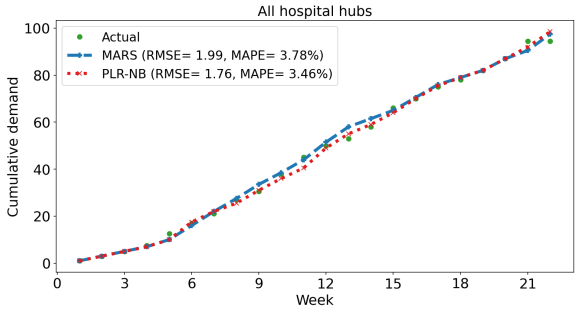
Figure 3.4: Model performance in forecasting CCP supply



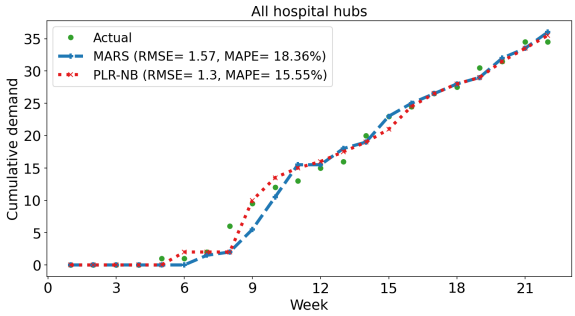
(a) Total units



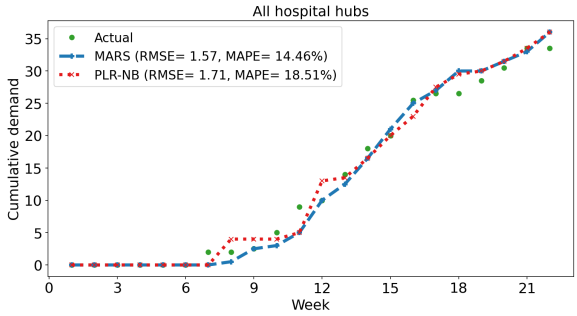
(b) A units



(c) O units



(d) B units



(e) AB units

Figure 3.5: Model performance in forecasting CCP demand

Hub	MARS				
	Total (%)	A (%)	O (%)	B (%)	AB (%)
Hospital Hub 1	16.07	17.05	19.73	20.45	22.73
Hospital Hub 2	12.87	18.79	15.50	17.82	17.86
Hospital Hub 3	11.32	19.41	12.59	19.89	18.67
Hospital Hub 4	21.70	25.45	53.45	16.75	16.67
Hospital Hub 5	7.31	15.26	11.52	22.35	17.36
Hospital Hub 6	15.12	13.86	14.60	22.66	14.94
Hospital Hub 7	27.91	25.58	27.12	-	-
All Hospital Hubs	3.67	6.45	3.78	18.36	14.46
CBS Distribution Site	6.12	6.01	9.90	7.59	18.00

Table 3.4: MAPE of supply and demand forecasts under MARS

Hub	PLR-NB				
	Total (%)	A (%)	O (%)	B (%)	AB (%)
Hospital Hub 1	25.41	25.00	25.36	27.27	36.36
Hospital Hub 2	9.68	12.94	10.77	15.50	32.58
Hospital Hub 3	12.32	14.42	14.51	18.18	20.88
Hospital Hub 4	25.02	23.85	61.74	27.53	31.82
Hospital Hub 5	7.18	16.66	10.54	26.89	17.06
Hospital Hub 6	13.55	9.31	12.24	25.77	21.29
Hospital Hub 7	30.52	56.28	31.82	-	-
All Hospital Hubs	4.83	6.27	3.46	15.55	18.51
CBS Distribution Site	5.42	5.21	11.33	11.40	17.73

Table 3.5: MAPE of supply and demand forecasts under PLR-NB

3.10, we analyze the sensitivity of our model to different allocation settings by forecasting the aggregate supply and demand under MARS and PLR-NB, respectively, and using the distribution of Canadian blood groups for calculating supply and demand for each resource. A total of $m = 300$ runs are considered for calculating $\overline{u}_T$ and $\overline{z}_T$ and the standard error (SE) of their corresponding 95% confidence intervals is reported.

			Identity		CONCOR-1	
Hub	Total Demand	Total Forecast Demand	$\overline{u}_T$	$\overline{z}_T$ (%)	$\overline{u}_T$	$\overline{z}_T$ (%)
Hospital Hub 1	12.5	12	1.00	8.00	1.00	8.00
Hospital Hub 2	29	30.5	0.50	1.72	1.00	3.45
Hospital Hub 3	74	79	4.50	6.08	10.00	13.51
Hospital Hub 4	26	30.5	1.00	3.85	1.50	5.77
Hospital Hub 5	30	30	1.50	5.00	1.00	3.33
Hospital Hub 6	105	110.5	8.50	8.10	10.00	9.52
Hospital Hub 7	17.5	20.5	0	0	0	0

Table 3.6: Resource allocation results under forecast supply and demand for each resource (MARS)

The key observations obtained from using our data-driven resource allocation model in the CONCOR-1 case study are as follows:

1. **The role of hospital hubs' proportion of CCP demand —** *A hospital hub's proportion of CCP demand for a resource can lead to its demand not being fully met if the available supply for the resource on a particular week is limited compared to its total demand for the same week. Our approach can help minimize the unmet CCP demand ratios and lead to balanced and fair CCP allocation decisions.*

We note that there is a shortage of 10 O, 5.5 B, and 7.5 AB CCP units due to the limitation in supply for these blood groups, as reported in Table 3.2. Thus, the CCP demand proportion of a hospital hub for a particular resource and the variance in the total demand between the hospital hubs in each week can affect $\overline{u}_T$ and $\overline{z}_T$. For instance, if the available

			Identity		CONCOR-1	
Hub	Total Demand	Total Forecast Demand	$\overline{u}_T$	$\overline{z}_T$ (%)	$\overline{u}_T$	$\overline{z}_T$ (%)
Hospital Hub 1	12.5	11.5	1.00	8.00	1.00	8.00
Hospital Hub 2	29	30.5	1.50	5.17	1.50	5.17
Hospital Hub 3	74	79	5.00	6.76	3.00	4.05
Hospital Hub 4	26	30.5	0.50	1.92	1.00	3.85
Hospital Hub 5	30	30.5	1.00	3.33	1.00	3.33
Hospital Hub 6	105	109.5	9.00	8.57	11.00	10.48
Hospital Hub 7	17.5	20.5	0	0	0.50	2.86

Table 3.7: Resource allocation results under forecast supply and demand for each resource (PLR-NB)

		MIP Allocations					
Hub	Total Demand	Identity		CONCOR-1		Allocations in CONCOR-1	
		$\overline{u}_T$	$\overline{z}_T$ (%)	$\overline{u}_T$	$\overline{z}_T$ (%)	$\overline{u}_T$	$\overline{z}_T$ (%)
Hospital Hub 1	12.5	1.50	12.00	1.00	8.00	3.00	24.00
Hospital Hub 2	29	1.00	3.45	0.50	1.72	1.50	5.17
Hospital Hub 3	74	7.00	9.46	4.00	5.41	20.00	27.02
Hospital Hub 4	26	2.00	7.69	0.50	1.92	4.50	17.31
Hospital Hub 5	30	1.50	5.00	0.50	1.67	5.00	16.67
Hospital Hub 6	105	10.00	9.52	7.00	6.67	27.00	25.71
Hospital Hub 7	17.5	0	0	0	0	1.00	5.71

Table 3.8: MIP model allocations under actual supply and demand versus actual allocations in CONCOR-1

supply on week t can meet all the demand on week t, a hospital hub's proportion of CCP demand for a resource is not an issue. On the other hand, if the available supply on week t is limited compared to the total demand for the same week, a hospital hub's high demand proportion for a limited resource CCP can lead to its demand not being fully met. This issue is unavoidable in our setting where we make predictions and allocations on a weekly basis and the optimization model finds a solution based only on the current situation.

We observe in Table 3.6 and Table 3.7 that Hospital Hub 1, Hospital Hub 3, and Hospital Hub 6 have larger $\overline{z}_T$ under both compatibility matrices. Hospital Hub 1's $\overline{z}_T$ is high because of its unmet CCP demand proportions. However, its $\overline{u}_T$ is not high and is due to the demand

			Identity		CONCOR-1	
Hub	Total Demand	Total Forecast Demand	$\overline{u}_T \pm SE$	$\overline{z}_T \pm SE$ (%)	$\overline{u}_T \pm SE$	$\overline{z}_T \pm SE$ (%)
Hospital Hub 1	12.5	10.5	1.00 ± 0.03	8.04 ± 0.26	0.93 ± 0.03	7.47 ± 0.27
Hospital Hub 2	29	30	1.62 ± 0.10	5.57 ± 0.36	1.65 ± 0.10	5.70 ± 0.34
Hospital Hub 3	74	77	3.66 ± 0.15	4.94 ± 0.20	5.97 ± 0.20	8.07 ± 0.27
Hospital Hub 4	26	28.5	1.29 ± 0.08	4.97 ± 0.31	1.52 ± 0.08	5.85 ± 0.30
Hospital Hub 5	30	29.5	1.26 ± 0.06	4.20 ± 0.19	1.21 ± 0.05	4.03 ± 0.16
Hospital Hub 6	105	110	8.57 ± 0.17	8.16 ± 0.16	7.63 ± 0.19	7.27 ± 0.18
Hospital Hub 7	17.5	20.5	0.02 ± 0.01	0.14 ± 0.07	0.42 ± 0.03	2.38 ± 0.15

Table 3.9: Resource allocation results under aggregate forecast supply and demand (MARS)

			Identity		CONCOR-1	
Hub	Total Demand	Total Forecast Demand	$\overline{u}_T \pm SE$	$\overline{z}_T \pm SE$ (%)	$\overline{u}_T \pm SE$	$\overline{z}_T \pm SE$ (%)
Hospital Hub 1	12.5	10.5	0.94 ± 0.04	7.49 ± 0.30	0.97 ± 0.04	7.73 ± 0.30
Hospital Hub 2	29	30	1.28 ± 0.07	4.41 ± 0.26	$1.42 pm 0.08$	4.89 ± 0.28
Hospital Hub 3	74	77	3.95 ± 0.19	5.34 ± 0.25	6.28 ± 0.20	8.48 ± 0.28
Hospital Hub 4	26	28.5	0.58 ± 0.06	2.24 ± 0.24	1.04 ± 0.07	4.02 ± 0.28
Hospital Hub 5	30	29.5	0.98 ± 0.03	3.28 ± 0.11	1.17 ± 0.05	3.89 ± 0.16
Hospital Hub 6	105	110	9.3 ± 0.14	8.86 ± 0.14	9.15 ± 0.29	8.71 ± 0.28
Hospital Hub 7	17.5	20.5	0.06 ± 0.02	0.31 ± 0.10	0.56 ± 0.04	3.17 ± 0.25

Table 3.10: Resource allocation results under aggregate forecast supply and demand (PLR-NB)

that was not met on the last week in our model (and may have been satisfied if we continued for additional weeks). In fact, the supply and demand for the last week can highly affect the final unmet demands of all hospital hubs. Hospital Hub 3 and Hospital Hub 6 require a higher proportion of B and AB CCP units compared to the other hospital hubs on weeks when the supply for these units is limited. A hospital hub's unmet demand on a particular week is moved to the next week and will have an effect on its future allocations. We observe that the rest of the hospital hubs have similar $\overline{u}_T$ and $\overline{z}_T$ values. This suggests that our proposed data-driven MIP model leads to a reasonable and fair balance of limited CCP products between the hospital hubs under both the MARS and PLR-NB forecasting models.

2. **The role of compatibility matrix**

2.1. *When the actual supply and demand is unbalanced, not known before allocation, and hence error due to forecasting is present, using the identity compatibility matrix in the MIP level is preferred as it prevents the allocation of limited resources to demand for a more abundant resource.*

We notice in Table 3.6 and Table 3.7 that the total $\overline{u}_T$ for the identity compatibility matrix is lower than for the CONCOR-1 compatibility matrix. While the CONCOR-1 compatibility matrix satisfies as much demand as possible in the current week, its use in this real-time setting can cause issues for future demand, in particular by allocating resources with lower long term supply to demands that can (eventually) be satisfied by more abundant resources. The identity compatibility matrix is hence preferred for situations with unbalanced demand and supply. We found that under both models and under the CONCOR-1 compatibility matrix, the shortage for O units was compensated by excess A units, and excess B demands were met by AB units. Since the supply for A units was the highest during the trial, and

the supply for AB units was the lowest, using AB units to meet demand from other blood groups led to shortages of AB units in future weeks. Furthermore, since the supply and demand forecasts are used instead of the actual values, we might overestimate the demand for a rare resource, such as the AB blood group CCP that is also compatible with another resource, so its allocation increases the $\overline{u}_T$ of some hospital hubs at the end. However, under both models and under the identity compatibility matrix, greedy allocations at the MIP level are prevented. If the forecasting error is not too large, the combination of the identity compatibility matrix at the MIP level and the CONCOR-1 compatibility matrix at the inventory level of hospital hubs is preferred under forecast supply and demand as the identity compatibility matrix at the MIP level prevents greedy allocations that cause issues for future demand, but the forecasting error results in the allocation of some compatible resources that combined with the CONCOR-1 compatibility matrix at the hospital hubs' inventory level can help meet more demand. Therefore, for our primary study, we recommend this combination of compatibility matrices that prevents future misallocation and absorbs the forecasting errors in an effective manner.

2.2. *The CONCOR-1 compatibility matrix results in lower unmet demand when the actual supply and demand for each week is known. The total unmet demand under this compatibility matrix is close to the lowest achievable value, the actual shortage in supply.*

While using the identity compatibility matrix in Table 3.6 and Table 3.7 where supply and demand forecasts are used exhibits better results, the opposite is true when solving the MIP model based on the assumption of knowing the actual supply and demand for each week. This can be observed by comparing the total $\overline{u}_T$ in Table 3.8 under "MIP Allocations" for both compatibility matrices (23 versus 13.5). When the actual supply and demand are used,

over allocation of limited resources will not happen and the final unmet demand is close to the actual supply shortage. We note again that 5.5 B, and 7.5 AB CCP units cannot be met due to supply shortage which is almost exactly the total $\overline{u}_T$ result of our MIP model under the CONCOR-1 compatibility matrix (13.5). This observation also reinforces the notion in the previous observation that forecasting errors (combined with unbalanced supply and demand) are what drive the recommendation of the identity compatibility matrix under forecast supply and demand. The identity compatibility matrix does not allow cross-transfusion, so 10 O units cannot be satisfied by the excess A units due to supply shortage. The total unmet demand under the identity compatibility matrix (23) further validates the performance of our model in efficiently allocating the available resources as it is equal to the actual shortage in supply, as shown in Table 3.2.

2.3. *The identity compatibility matrix results in lower unmet demand under forecast supply and demand compared to when it is used under actual supply and demand, as reasonable forecast error can better help in meeting the demand.*

Comparing the $\overline{z}_T$ values in Table 3.6 and Table 3.7 with Table 3.8 under the CONCOR-1 compatibility matrix for each hospital hub, we observe that $\overline{z}_T$ for all hospital hubs is improved (or is the same) in Table 3.8, significantly so for Hospital Hub 3 and Hospital Hub 6. Furthermore, the $\overline{z}_T$ values in Table 3.8 are closer to each other under the CONCOR-1 compatibility matrix than those reported in Table 3.6 and Table 3.7, where the results are affected by the forecasting errors. However, we notice that all $\overline{z}_T$ values in Table 3.6 and Table 3.7 under the identity compatibility matrix are lower than (or the same as) those in Table 3.8. The reason is that using the identity compatibility matrix at the MIP level under actual supply and demand is not a good choice considering that the MIP model never assigns CCP units more than the actual demand. This choice leads to the CONCOR-1 compatibility

matrix at the inventory level of the hospital hubs not helping at all in meeting the demand. However, under forecast supply and demand, although the identity compatibility matrix prevents the allocation of a resource to a demand for another resource, the forecasts induce some compatible assignments. Hence, when the identity compatibility matrix at the MIP level is combined with the CONCOR-1 compatibility matrix at the inventory level of the hospital hubs, and the supply and demand forecasts do not have large errors, there is more chance to meet the demand, as what is observed for Hospital Hub 3 and Hospital Hub 6.

2.4. *The MIP model's results after using the aggregate forecast supply and demand (sensitivity analysis) are close to the primary study when the identity compatibility matrix is used, and improved under the CONCOR-1 compatibility matrix. However, the identity compatibility matrix is a better choice than the CONCOR-1 compatibility matrix when using the aggregate forecast supply and demand and in the presence of forecast errors, consistent with what is observed in the primary study.*

We notice that although the results under the identity compatibility matrix remain almost unchanged under both models, the total $\overline{u}_T$ and $\overline{z}_T$ values under the CONCOR-1 compatibility matrix in Table 3.9 and Table 3.10 are lower than those in Table 3.6 and Table 3.7. This appears to arise due to the fact that including randomness for generating the supply and demand for each blood group based on the aggregate supply and demand may slightly lower the forecasting error. Furthermore, performing sufficient runs helps to smooth the effect of scenarios where resources with limited supply, such as the AB blood group CCP, are allocated in a greedy manner and cause issues for future demand. We note that the results under the identity matrix are better than those under the CONCOR-1 compatibility matrix in Table 3.9 and Table 3.10 as the identity compatibility matrix prevents any cross-allocation of units and lowers the effect of forecast errors, consistent with what we observed in Table 3.6

and Table 3.7. Comparing the $\overline{u}_T$ and $\overline{z}_T$ values in Table 3.9, Table 3.10, and Table 3.8 under "MIP Allocations", it is important to note that while having exact knowledge of supply and demand leads to a fairer allocation, the degradation in performance is not unreasonable if we use the forecast values instead, further supporting the efficacy of our approach. However, as previously discussed, using the identity compatibility matrix in the MIP model under actual supply and demand leads to more unmet demand (as compared to using the CONCOR-1 compatibility matrix) due to not allocating excess units of compatible resources, which prevents the CONCOR-1 compatibility matrix at the inventory level of the hospital hubs to help meet the demand. Similar to our primary study, if the forecast errors are not too large, using the identity compatibility matrix at the MIP level under forecast supply and demand results in making some compatible assignments that can better help meet the demand when combined with the CONCOR-1 compatibility matrix at the hospital hubs' inventory level.

For this case study and based on all the above observations, we conclude that under both forecasting models, our primary study when the identity compatibility matrix is used at the MIP level is the most promising choice for our real-time setting. The reason is that as long as the forecast errors for the individual blood groups are not too large, using the combination of the identity compatibility matrix at the MIP level and the CONCOR-1 compatibility matrix at the inventory level helps in both preventing greedy allocations and compensating for forecasting errors. Furthermore, no randomness due to supply and demand probabilities is included in this choice for calculating the supply and demand for each blood group (as what is assumed in our sensitivity analysis), and the assignments of the model are not affected by any ABO blood distributions.

3. **The role of the MIP model —** *The results obtained from our MIP model both under actual and forecast supply are preferable to what was used in practice.*

We compare $\overline{u}_T$ and $\overline{z}_T$ for both compatibility matrices using the MIP model allocations under actual supply and demand, and the actual allocations in the CONCOR-1 trial. We observe that the results under "MIP Allocations" are more fair than those under "Allocations in CONCOR-1" (the $\overline{z}_T$ values are closer to each other). Furthermore, the total $\overline{u}_T$ under "MIP Allocations" is notably lower (23 and 13.5 versus 62) which further supports the use of our optimization model, as it accounts for the hospital hubs' unmet demand and inventories.

3.5 Conclusion and Future Work

Decision-makers often face challenges in terms of (i) allocating limited resources, such as vaccines, blood products, and medical equipment, and (ii) forecasting the supply and demand for these resources during epidemics as there is limited knowledge and historical data about disease demographics. This work has considered the problem of real-time short-term supply and demand forecasting and fair allocation of limited resources during epidemics by using PLR forecasting models and introducing a data-driven MIP resource allocation model. We have studied the application of our proposed MIP model in a CCP clinical trial case study with the objective of minimizing each hospital hub's unmet ratio of CCP demand. We showed that as long as a hospital hub does not have a high demand proportion for a limited blood group CCP on a particular week (in which case a fair allocation is not possible), our MIP model leads to a balanced and fair final unmet ratio of CCP demand between the hospital hubs under both forecasting models. We also showed that allocating a compatible resource to satisfy the demand for a resource helps in situations where the actual supply and demand is known, but might be problematic when we are forecasting the supply and demand and the actual supply of the compatible resource is limited. It would be interesting to investigate the range of the forecasting errors within which a particular compatibility

matrix is preferred.

Examining PLR forecasting models on larger datasets and comparing their performance with more advanced machine learning and time-series forecasting models could be investigated in future work. We have addressed several challenges that arise when dealing with sparse data in a real-time setting. We are interested in evaluating other forecasting models that can adapt to these challenges while yielding reasonable forecasts. It would be worthwhile to investigate our MIP model's performance when other supply and demand forecasting models are used.

Finally, multiple objective functions and different notions of fairness can be considered in a single resource allocation problem. Using more than one objective function and focusing on other notions of fairness such as minimizing the aggregate unmet demand over all hospital hubs or minimizing the transportation costs of shipping CCP units with respect to the location of the hospital hubs are examples of problems of interest. It would also be interesting to see our MIP model's performance when applied to other allocation settings with limited supply. Our MIP model only considers a centralized supplier, thus we leave the model's evaluation with the presence of multiple suppliers to future investigation.

Acknowledgement

We would like to thank Julie Carruthers, Erin Jamula, and Melanie St John at the McMaster Centre for Transfusion Research for their administrative support. We also acknowledge the support and funding provided by a Mitacs Research Training Award (Award IT22358), the McMaster Centre for Transfusion Research, and the Natural Sciences and Engineering Research Council of Canada (NSERC) Discovery Grant Program (RGPIN-2016-04518).

Bibliography

1. A. Kumar and J. Kleinberg, "Fairness measures for resource allocation," in *Proceedings 41st Annual Symposium on Foundations of Computer Science*. IEEE, 2000, pp. 75–85.

2. Ö. Karsu and H. Erkan, "Balance in resource allocation problems: A changing reference approach," *OR Spectrum*, vol. 42, no. 1, pp. 297–326, 2020.

3. Ö. Karsu and A. Morton, "Incorporating balance concerns in resource allocation decisions: A bi-criteria modelling approach," *Omega*, vol. 44, pp. 70–82, 2014.

4. H. K. Smith, P. R. Harper, and C. N. Potts, "Bicriteria efficiency/equity hierarchical location models for public service application," *Journal of the Operational Research Society*, vol. 64, no. 4, pp. 500–512, 2013.

5. A. M. Mestre, M. D. Oliveira, and A. Barbosa-Póvoa, "Organizing hospitals into networks: A hierarchical and multiservice model to define location, supply and referrals in planned hospital systems," *OR Spectrum*, vol. 34, no. 2, pp. 319–348, 2012.

6. H. Heitmann and W. Brüggemann, "Preference-based assignment of university students to multiple teaching groups," *OR Spectrum*, vol. 36, no. 3, pp. 607–629, 2014.

7. Ö. Karsu and A. Morton, "Inequity averse optimization in operational research," *European Journal of Operational Research*, vol. 245, no. 2, pp. 343–359, 2015.

8. C. S. Kraft, A. L. Hewlett, S. Koepsell, A. M. Winkler, C. J. Kratochvil, L. Larson, J. B. Varkey, A. K. Mehta, G. M. Lyon III, R. J. Friedman-Moraco *et al.*, "The use of

TKM-100802 and convalescent plasma in 2 patients with Ebola virus disease in the United States," *Clinical Infectious Diseases*, vol. 61, no. 4, pp. 496–502, 2015.

9. J. Van Griensven, T. Edwards, X. de Lamballerie, M. G. Semple, P. Gallian, S. Baize, P. W. Horby, H. Raoul, N. Magassouba, A. Antierens *et al.*, "Evaluation of convalescent plasma for Ebola virus disease in Guinea," *New England Journal of Medicine*, vol. 374, no. 1, pp. 33–42, 2016.

10. I. F. Hung, K. K. To, C.-K. Lee, K.-L. Lee, K. Chan, W.-W. Yan, R. Liu, C.-L. Watt, W.-M. Chan, K.-Y. Lai *et al.*, "Convalescent plasma treatment reduced mortality in patients with severe pandemic influenza A (H1N1) 2009 virus infection," *Clinical Infectious Diseases*, vol. 52, no. 4, pp. 447–456, 2011.

11. B. Zhou, N. Zhong, and Y. Guan, "Treatment with convalescent plasma for influenza A (H5N1) infection," *New England Journal of Medicine*, vol. 357, no. 14, pp. 1450–1451, 2007.

12. L. Chen, J. Xiong, L. Bao, and Y. Shi, "Convalescent plasma as a potential therapy for COVID-19," *The Lancet Infectious Diseases*, vol. 20, no. 4, pp. 398–400, 2020.

13. F. Brauer, "Compartmental models in epidemiology," in *Mathematical Epidemiology*. Springer, 2008, pp. 19–79.

14. T. P. Velavan and C. G. Meyer, "The COVID-19 epidemic," *Tropical Medicine & International Health*, vol. 25, no. 3, p. 278, 2020.

15. A. Tomar and N. Gupta, "Prediction for the spread of COVID-19 in India and effectiveness of preventive measures," *Science of the Total Environment*, vol. 728, p. 138762, 2020.

16. J. Gong, J. Ou, X. Qiu, Y. Jie, Y. Chen, L. Yuan, J. Cao, M. Tan, W. Xu, F. Zheng *et al.*, "A tool for early prediction of severe coronavirus disease 2019 (COVID-19): A multicenter study using the risk nomogram in Wuhan and Guangdong, China," *Clinical Infectious Diseases*, vol. 71, no. 15, pp. 833–840, 2020.

17. M. E. Chowdhury, T. Rahman, A. Khandakar, S. Al-Madeed, S. M. Zughaier, H. Hassen, M. T. Islam *et al.*, "An early warning tool for predicting mortality risk of COVID-19 patients using machine learning," *arXiv preprint arXiv:2007.15559*, 2020.

18. G. E. Weissman, A. Crane-Droesch, C. Chivers, T. Luong, A. Hanish, M. Z. Levy, J. Lubken, M. Becker, M. E. Draugelis, G. L. Anesi *et al.*, "Locally informed simulation to predict hospital capacity needs during the COVID-19 pandemic," *Annals of Internal Medicine*, vol. 173, no. 1, pp. 21–28, 2020.

19. S. P. Silal, F. Little, K. I. Barnes, and L. J. White, "Sensitivity to model structure: A comparison of compartmental models in epidemiology," *Health Systems*, vol. 5, no. 3, pp. 178–191, 2016.

20. J. O. Ferstad, A. J. Gu, R. Y. Lee, I. Thapa, A. Y. Shin, J. A. Salomon, P. Glynn, N. H. Shah, A. Milstein, K. Schulman *et al.*, "A model to forecast regional demand for COVID-19 related hospital beds," *medRxiv*, 2020.

21. K. Nikolopoulos, S. Punia, A. Schäfers, C. Tsinopoulos, and C. Vasilakis, "Forecasting and planning during a pandemic: COVID-19 growth rates, supply chain disruptions, and governmental decisions," *European Journal of Operational Research*, vol. 290, no. 1, pp. 99–115, 2021.

22. N. Li, F. Chiang, D. G. Down, and N. M. Heddle, "A decision integration strategy

for short-term demand forecasting and ordering for red blood cell components," *Operations Research for Health Care*, p. 100290, 2021.

23. D. B. Suits, A. Mason, and L. Chan, "Spline functions fitted by standard regression methods," *The Review of Economics and Statistics*, pp. 132–139, 1978.

24. J. B. Rosen and P. M. Pardalos, "Global minimization of large-scale constrained concave quadratic problems by separable programming," *Mathematical Programming*, vol. 34, no. 2, pp. 163–174, 1986.

25. B. Strikholm, "Determining the number of breaks in a piecewise linear regression model," SSE/EFI Working Paper Series in Economics and Finance, Tech. Rep., 2006.

26. L. Yang, S. Liu, S. Tsoka, and L. G. Papageorgiou, "Mathematical programming for piecewise linear regression analysis," *Expert Systems with Applications*, vol. 44, pp. 156–167, 2016.

27. T. Hong, M. Gui, M. E. Baran, and H. L. Willis, "Modeling and forecasting hourly electric load by multiple linear regression with interactions," in *IEEE PES General Meeting*. IEEE, 2010, pp. 1–8.

28. C.-Y. Huang and G.-H. Tzeng, "Multiple generation product life cycle predictions using a novel two-stage fuzzy piecewise regression analysis method," *Technological Forecasting and Social Change*, vol. 75, no. 1, pp. 12–31, 2008.

29. P.-C. Chang, C.-Y. Fan, and C.-H. Liu, "Integrating a piecewise linear representation method and a neural network model for stock trading points prediction," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 39, no. 1, pp. 80–92, 2008.

30. N. F. Butte, W. W. Wong, A. L. Adolph, M. R. Puyau, F. A. Vohra, and I. F. Zakeri, "Validation of cross-sectional time series and multivariate adaptive regression splines models for the prediction of energy expenditure in children and adolescents using doubly labeled water," *The Journal of Nutrition*, vol. 140, no. 8, pp. 1516–1523, 2010.

31. S.-M. Chou, T.-S. Lee, Y. E. Shao, and I.-F. Chen, "Mining the breast cancer pattern using artificial neural networks and multivariate adaptive regression splines," *Expert Systems with Applications*, vol. 27, no. 1, pp. 133–142, 2004.

32. D. Yao, J. Yang, and X. Zhan, "A novel method for disease prediction: Hybrid of random forest and multivariate adaptive regression splines," *Journal of Computers*, vol. 8, no. 1, pp. 170–177, 2013.

33. D. Senthilkumar and S. Paulraj, "Diabetes disease diagnosis using multivariate adaptive regression splines," *AGE*, vol. 768, p. 52, 2013.

34. J. E. S. Lasheras, C. G. Donquiles, P. J. G. Nieto, J. J. J. Moleon, D. Salas, S. L. S. Gómez, A. J. M. de la Torre, J. González-Nuevo, L. Bonavera, J. C. Landeira *et al.*, "A methodology for detecting relevant single nucleotide polymorphism in prostate cancer with multivariate adaptive regression splines and backpropagation artificial neural networks," *Neural Computing and Applications*, vol. 32, no. 5, pp. 1231–1238, 2020.

35. N. B. Serrano, A. S. Sánchez, F. S. Lasheras, F. J. Iglesias-Rodríguez, and G. F. Valverde, "Identification of gender differences in the factors influencing shoulders, neck and upper limb MSD by means of multivariate adaptive regression splines (MARS)," *Applied Ergonomics*, vol. 82, p. 102981, 2020.

36. J.-M. López-Lozano, T. Lawes, C. Nebot, A. Beyaert, X. Bertrand, D. Hocquet, M. Aldeyab, M. Scott, G. Conlon-Bingham, D. Farren *et al.*, "A nonlinear time-series analysis approach to identify thresholds in associations between population antibiotic use and rates of resistance," *Nature Microbiology*, vol. 4, no. 7, pp. 1160–1172, 2019.

37. C. Katris, "A time series-based statistical approach for outbreak spread forecasting: Application of COVID-19 in Greece," *Expert Systems with Applications*, vol. 166, p. 114077, 2021.

38. M. Liu and J. Liang, "Dynamic optimization model for allocating medical resources in epidemic controlling," *Journal of Industrial Engineering and Management (JIEM)*, vol. 6, no. 1, pp. 73–88, 2013.

39. V. M. Preciado, M. Zargham, C. Enyioha, A. Jadbabaie, and G. Pappas, "Optimal vaccine allocation to control epidemic outbreaks in arbitrary networks," in *52nd IEEE Conference on Decision and Control*. IEEE, 2013, pp. 7486–7491.

40. H. Yarmand, J. S. Ivy, B. Denton, and A. L. Lloyd, "Optimal two-phase vaccine allocation to geographically different regions under uncertainty," *European Journal of Operational Research*, vol. 233, no. 1, pp. 208–219, 2014.

41. V. M. Preciado, M. Zargham, C. Enyioha, A. Jadbabaie, and G. J. Pappas, "Optimal resource allocation for network protection against spreading processes," *IEEE Transactions on Control of Network Systems*, vol. 1, no. 1, pp. 99–108, 2014.

42. S. Han, V. M. Preciado, C. Nowzari, and G. J. Pappas, "Data-driven network resource allocation for controlling spreading processes," *IEEE Transactions on Network Science and Engineering*, vol. 2, no. 4, pp. 127–138, 2015.

43. N. P. Rachaniotis, T. K. Dasaklis, and C. P. Pappis, "A deterministic resource scheduling model in epidemic control: A case study," *European Journal of Operational Research*, vol. 216, no. 1, pp. 225–231, 2012.

44. N. Rachaniotis, T. K. Dasaklis, and C. Pappis, "Controlling infectious disease outbreaks: A deterministic allocation-scheduling model with multiple discrete resources," *Journal of Systems Science and Systems Engineering*, vol. 26, no. 2, pp. 219–239, 2017.

45. L. Sun, G. W. DePuy, and G. W. Evans, "Multi-objective optimization models for patient allocation during a pandemic influenza outbreak," *Computers & Operations Research*, vol. 51, pp. 350–359, 2014.

46. A. Anparasan and M. Lejeune, "Resource deployment and donation allocation for epidemic outbreaks," *Annals of Operations Research*, vol. 283, no. 1, pp. 9–32, 2019.

47. M. Du, A. Sai, and N. Kong, "A data-driven optimization approach for multi-period resource allocation in cholera outbreak control," *European Journal of Operational Research*, vol. 291, no. 3, pp. 1106–1116, 2021.

48. R. Bekker, M. Uit Het Broek, and G. Koole, "Modeling COVID-19 hospital admissions and occupancy in the Netherlands," *arXiv preprint arXiv:2102.11021*, 2021.

49. P. Ellaway, "Cumulative sum technique and its application to the analysis of peristimulus time histograms," *Electroencephalography and Clinical Neurophysiology*, vol. 45, no. 2, pp. 302–304, 1978.

50. J. H. Friedman, "Multivariate adaptive regression splines," *The Annals of Statistics*, pp. 1–67, 1991.

51. N. Golovchenko, "Least-squares fit of a continuous piecewise linear function," http://www.golovchenko.org/docs/ContinuousPiecewiseLinearFit.pdf, 2004, [Online; accessed 26-June-2021].

52. J. E. Jarrett and S. B. Khumuwala, "A study of forecast error and covariant time series to improve forecasting for financial decision making," *Managerial Finance*, vol. 13, no. 2, pp. 20–24, 1987.

53. M. Lu and Y. Chen, "Improved estimation and forecasting through residual-based model error quantification," *SPE Journal*, vol. 25, no. 02, pp. 951–968, 2020.

54. K. L. Chai, S. J. Valk, V. Piechotta, C. Kimber, I. Monsef, C. Doree, E. M. Wood, A. A. Lamikanra, D. J. Roberts, Z. McQuilten *et al.*, "Convalescent plasma or hyperimmune immunoglobulin for people with COVID-19: A living systematic review," *Cochrane Database of Systematic Reviews*, no. 10, 2020.

55. P. Bégin, J. Callum, N. Heddle, R. Cook, M. P. Zeller, A. Tinmouth, D. Fergusson, M. M. Cushing, M. J. Glesby, M. Chassé *et al.*, "Convalescent plasma for adults with acute COVID-19 respiratory illness (CONCOR-1): Study protocol for an international, multicenter, randomized, open-label trial," *Trials*, vol. 22, no. 323, 2021.

56. E. M. Bloch, R. Goel, S. Wendel, T. Burnouf, A. Z. Al-Riyami, A. L. Ang, V. DeAngelis, L. J. Dumont, K. Land, C.-k. Lee *et al.*, "Guidance for the procurement of COVID-19 convalescent plasma: Differences between high- and low-middle-income countries," *Vox Sanguinis*, vol. 116, no. 1, pp. 18–35, 2021.

57. Canadian Blood Services, "What's my blood type?" http://www.blood.ca/en/blood/donating-blood/whats-my-blood-type, 2021, [Online; accessed 26-June-2021].

58. J. Rudy, "py-earth: A Python implementation of Jerome Friedman's multivariate adaptive regression splines," http://www.github.com/scikit-learn-contrib/py-earth, 2016, [Online; accessed 26-June-2021].

59. M. Zietz, J. Zucker, and N. P. Tatonetti, "Associations between blood type and COVID-19 infection, intubation, and death," *Nature Communications*, vol. 11, no. 1, pp. 1–6, 2020.

Chapter 4

Conclusion

Resource management is challenging when exact information about the supply and/or demand for resources is not known before making resource allocation decisions. Many other factors can affect this process such as limited historical information about the resources, limited supply and heterogeneous demand, and the necessity of real-time or timely allocations. Thus, decision makers often use supply and/or demand estimates to allocate the available resources more efficiently. This thesis studied two problems in resource management that are similar in the sense that they consider scenarios where there are high and heterogeneous demand for a limited resource, exact information about the supply and/or demand is not available within the decision making period, and a timely resource allocation is preferred. Both problems considered scenarios where only estimates of the demand is known and the second problem also considered estimating the supply for the resources.

In the first problem, we studied scheduling a single-server system with job processing time estimates. We have introduced SEH, a novel heuristic that combines the merits of two size-based scheduling policies and requires minimal calculation overhead and no information about the jobs, rather than their size estimates. Our numerical results demonstrated that our

heuristic has desirable performance in minimizing both the MST and mean slowdown of the system when jobs exhibit estimation error distribution variance that is seen in practical settings. In future, we are interested in performing more investigations on how far SEH is from optimal and examining this policy under various estimation error models. It would also be worthwhile to examine the performance of policies that provide worst case performance guarantees and compare them to the performance of SEH. Finally, studying the extension and evaluation of SEH in multi-server systems in the presense of inexact job sizes is another candidate for future work.

The second problem tackles the issue of real-time short-term supply and demand forecasting and fair allocation of limited resources during emerging epidemics. We assumed that we have no historical data and the available data is very sparse and limited, which is observed during emerging epidemics. Furthermore, we did not rely on demographic information and epidemiological models, which require indeterminate parameters (such as location and time-specific parameters) and a periodic update of the parameters. The challenges that may arise in an online setting due to extrapolation and sparse data were addressed by suggesting potential solutions. We studied the application of our data-driven model in a CCP clinical case study and provided numerical results of the performance of our model in fairly allocating the CCP units and minimizing the unmet demand ratio for each of the participating hospital hubs. In future, we are interested in examining our model on larger datasets and compare our model forecasts with those from more advanced machine-learning and time-series models that can adapt to the mentioned challenges. Furthermore, investigating the use of multiple objective functions, focusing on other notions of fairness, considering multiple suppliers, and deploying our model to other allocation settings with limited supply is of interest.