MAINTAINING FAIRNESS IN STOCHASTIC CHEMOTHERAPY SCHEDULING

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR
THE DEGREE OF
MASTER OF SCIENCE
IN
INDUSTRIAL ENGINEERING

By
Batuhan Çelik
June 2024

MAINTAINING FAIRNESS IN STOCHASTIC CHEMOTHERAPY SCHEDULING
By Batuhan Çelik
June 2024

We certify that we have read this thesis and that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Özlem Karsu (Advisor)

Serhat Gül (Co-Advisor)

Melih Çelik

Bahar Çavdar

Approved for the Graduate School of Engineering and Science:

Orhan Arıkan
Director of the Graduate School

ABSTRACT

MAINTAINING FAIRNESS IN STOCHASTIC CHEMOTHERAPY SCHEDULING

Batuhan Çelik
M.S. in Industrial Engineering
Advisor: Özlem Karsu
Co-Advisor: Serhat Gül
June 2024

Chemotherapy scheduling is hard to manage under uncertainty in infusion durations, and focusing on expected performance measure values may lead to unfavorable outcomes for some patients. We aim to design daily patient appointment schedules considering fairness regarding patient waiting times. We propose using a metric that encourages fairness and efficiency in waiting time allocations. To optimize this metric, we formulate a two-stage stochastic mixed-integer nonlinear programming model. We employ a binary search algorithm to identify an optimal schedule, and then propose a modified binary search algorithm (MBSA) to enhance computational capability. Moreover, to address stochastic feasibility problems at each MBSA iteration, we introduce a novel reduce-and-augment algorithm that utilizes scenario set reduction and augmentation methods. We use real data from a major oncology hospital to show the efficacy of MBSA. We compare the schedules identified by MBSA with both the baseline schedules from the oncology hospital and those generated by commonly employed scheduling heuristics. We also compare our metric with a well-known inequity metric (the Gini coefficient) and a Rawlsian-type welfare function. Finally, we highlight the significance of considering uncertainty in infusion durations to maintain fairness while creating appointment schedules.

Keywords: healthcare operations, chemotherapy, scheduling, fairness, stochastic programming.

ÖZET

RASSAL KEMOTERAPİ ÇİZELGELEMESİNDE ADİLLİĞİN GÖZETİLMESİ

Batuhan Çelik
Endüstri Mühendisliği, Yüksek Lisans
Tez Danışmanı: Özlem Karsu
İkinci Tez Danışmanı: Serhat Gül
Haziran 2024

Kemoterapi randevu planlaması, infüzyon sürelerindeki belirsizlik altında yönetilmesi zor olan bir konudur ve sadece beklenen performans ölçü değerlerine odaklanmak, bazı hastalar için istenmeyen sonuçlara yol açabilir. Bu çalışma, hasta bekleme süreleri açısından adilliği dikkate alarak günlük hasta randevularını planlamayı amaçlamaktadır. Bu doğrultuda, verimlilik ve adilliği aynı anda gözeten özgün bir ölçek önerilmiş ve iki aşamalı rassal karmaşık tamsayılı doğrusal olmayan programlama modeli içinde ençoklanmıştır. Oluşturulan model için ikili arama algoritması geliştirilmiştir. Bu algoritmanın her iterasyonunda oluşturulan rassal olurluluk problemlerini çözmek için, yeni bir azalt-ve-artır algoritması önerilmiştir. Senaryo azaltma yaklaşımı, literatürden gelen senaryo azaltma teknikleri ile bir onkoloji hastanesinden gerçek veriler kullanılarak karşılaştırılmıştır. Algoritma ayrıca yaygın olarak kullanılan bir doğrusal olmayan programlama çözücüsü ile karşılaştırılmştır. Elde edilen çizelgeler problemin baz alındığı hastanenin mevcut uygulaması ve yaygın olarak kullanılan sezgisel sıralama yöntemleri ile oluşturulan çizelgelerle kıyaslanmış ve önerilen çizelgelerin daha iyi performans gösterdiği gözlemlenmiştir. Ayrıca, önerilen metriğin bilinen bir eşitsizlik metriğinden (Gini katsayısı) ve Rawls tipi bir refah fonksiyonundan farkı incelenmiştir. Son olarak stokastik çözümün değeri hesaplanarak infüzyon sürelerindeki belirsizliğin önemi vurgulanmıştır.

Anahtar sözcükler: sağlık hizmetleri operasyonları, kemoterapi, çizelgeleme, adillik, rassal programlama.

Acknowledgement

First of all, I would like to express my sincere gratitude to my advisors, Assoc. Prof. Özlem Karsu and Assist. Prof. Serhat Gül for unmeasurable support and guidance. Their dedication and attentive mentoring were instrumental in completing my thesis with pride and excitement. I have learned so much from them, and I am also grateful to have had the opportunity to work with them.

I would like to thank the jury members of my thesis committee, Assoc. Prof. Melih Çelik and Assist. Prof. Bahar Çavdar for reading my thesis with interest and giving valuable comments.

I would like to express my deepest appreciation to my family for their patience and support during my master's education, like any other journey I have been through. I would like to thank my mother, Hatice, my father, Ahmet, for their trust in me, and my brother, Berke Han, for challenging me to get better. Without their support I wouldn't be able to walk on this path.

I am fortunate to have wonderful friends to whom I owe much gratitude. Special thanks to Sarp Bora İlhan, the best part of Bilkent I found. I have collected an ocean of memories with him, and the time spent with him eases this difficult journey. I'm grateful to have Eminenur Gürlek, Ece Gündem and Çağan Kutay Er for supporting me wherever they are. I cannot find better friends.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

There will be more than 2 million new cases of cancer diagnosed in the United States in 2024, according to recent projections. However, despite the increase in cancer prevalence, the mortality rate due to cancer has dropped by 33% from 1991 to 2021 [1]. This significant decline can largely be attributed to advances in cancer treatment, which have led to an increased demand for outpatient chemotherapy ([2], [1]). The growing demand for outpatient chemotherapy clinics (OCCs) has made it challenging for OCC directors to manage, therefore effective approaches should be developed to create appointment schedules that meet the needs of both providers and patients.

To create an efficient appointment schedule for a given day in an OCC, it is crucial for the manager to carefully coordinate the resources required for *premedication* and *infusion* activities, as they constitute the primary phases of patient treatment. Prior to starting these activities, a patient is first seated on a chair. The premedication process involves setting up a patient and administering drugs to prevent potential side effects such as nausea and vomiting. Once the premedication is completed, the infusion process begins, where the chemotherapy drugs are delivered to the veins of the patient through an intravenous bag and catheter. During the infusion process, controlling the patient and maintaining the precise flow of medication is sufficient for the nurse. This allows the nurse to efficiently manage multiple patients receiving treatment in chairs. However, the premedication process requires intensive nursing assistance, and therefore, a nurse can only attend to the premedication of a single patient at a time.

The assessment of schedules in the OCC involves the consideration of conflicting metrics: nurse overtime and patient waiting time. To decrease operating costs and boost nurse job satisfaction, it is essential to limit overtime. Reducing patient waiting time is crucial to improve patient satisfaction levels. However, creating better schedules to achieve both goals is a daunting task due to uncertain infusion durations. Predicting these durations can be challenging because chemotherapy drugs can damage healthy cells and cause side effects, leading to early termination of treatment, slower administration, or changes in medication plans. Such deviations from anticipated infusion durations can increase patient waiting time if durations are underestimated during schedule creation or increase nurse overtime if they are overestimated.

Expected waiting time is a commonly used criterion for evaluating patient satisfaction level while accounting for uncertainty in infusion durations. However, relying solely on this criterion can cause a notable disparity in waiting times among patients and lead to inequitable chemotherapy schedules. Therefore, it is crucial to consider an alternative criterion that can ensure fairness in schedules to a certain degree and enhance patient experiences. To demonstrate the need for a new criterion, we compare the patient waiting times associated with two different schedules. We present the results for these two schedules in Tables 1a and 1b, respectively. The tables report waiting times for four patients across four infusion duration scenarios with equal probability of occurrence. Despite the fact that the second schedule results in a higher total expected waiting time, the waiting times of patients are within a narrower range. Table 1a shows that the largest difference between the waiting times of two patients for a given scenario is 100 minutes, while in Table 1b, this difference is reduced to 50 minutes.

Preventing the occurrence of excessive waiting time in outpatient clinics is important because research suggests that the level of unpleasantness experienced while waiting does not escalate proportionately with the duration of the waiting

Table 1.1: Waiting times for "hypothetical example #1" with 4 patients and 4 scenarios

(a) Schedule 1

	Patient				Average
Scenario	1	2	3	4	
1	100	0	0	0	25
2	30	0	0	0	7.5
3	0	0	10	0	2.5
4	0	0	0	30	7.5
Expected Average					10.62

(b) Schedule 2

	Patient				Average
Scenario	1	2	3	4	
1	20	0	10	50	20
2	20	0	10	10	10
3	0	0	10	10	5
4	20	0	5	10	8.75
Expected Average					10.93

period ([3],[4]). Patients generally find waiting acceptable up to a certain limit ([5], [6]). According to Huang (1994) [7], outpatient clinic patients tend to be reasonably satisfied with their waiting times, as long as it does not exceed 37 minutes. Patients may have even a higher waiting time threshold in situations where they are receiving critical medical care. For instance, Lau et al. (2014) [8] surveyed chemotherapy patients who visited their oncologist on the day of treatment and found that 83% of patients were willing to wait for more than 30 minutes for chemotherapy treatment after the oncologist appointment.

In this study, we aim to prevent excessive waiting time and hence ensure equitable scheduling of chemotherapy appointments. The motivation for incorporating fairness into chemotherapy scheduling arises from our observations at a chemotherapy unit. For example, waiting time histograms from two randomly chosen days in our dataset, available in the online appendix, indicate an unbalanced distribution of waiting times. This issue is further supported by discussions with oncology unit staff, who consistently witness unbalanced waiting times across patients. To achieve our goal, we propose a metric encouraging fairness and efficiency and incorporate it into a two-stage stochastic mixed-integer nonlinear programming (SMINP) model. To the best of our knowledge, our study is the first incorporating fairness in a chemotherapy scheduling setting. Using the SMINP model, we sequence patients and set appointment times under uncertainty in infusion durations for a given daily list of patients. We also assign patients to nurses and chairs based on an optimal myopic policy. We maximize our proposed metric in the objective function while imposing a restriction on nurse overtime.

Our approach to addressing the nonlinear constraints in the SMINP model involves using a modified version of the binary search algorithm (MBSA). While this algorithm effectively omits these constraints, it then solves stochastic feasibility problems at each iteration. To solve these problems, we introduce a novel *reduce-and-augment* algorithm. This approach first reduces the original scenario set to a smaller size and then iteratively augments the scenario set until a feasible solution is found. We test multiple variants of scenario reduction algorithms for the first step of the reduce-and-augment algorithm. To test the effectiveness of the MBSA, we compare it with a commercial nonlinear programming solver. Additionally, we analyze the MBSA solutions and compare them with the baseline schedules and those obtained by commonly employed scheduling heuristics. Finally, we measure the value of the stochastic solution.

Chapter 2

Literature Review

The literature review is composed of three distinct sets of articles. The first set pertains to chemotherapy scheduling, while the second and third sets focus on robust optimization and fairness issues in general outpatient scheduling, respectively. For a comprehensive review of chemotherapy scheduling studies, we refer the reader to the works of Lam‘e et al. (2016) [9] and Hadid et al. (2022) [10].

2.1 Chemotherapy scheduling

The chemotherapy scheduling studies in the literature can be categorized based on the setting considered as deterministic and stochastic. The ones considering deterministic problems can be further classified based on the decisions made: some studies study planning and scheduling decisions while others solely focus on scheduling.

Turkcan et al. (2012) [11] formulate two separate models for planning and appointment scheduling. The planning model considers resource utilization while the scheduling model assigns the patients to slots to minimize completion time. Hooshangi-Tabrizi et al. (2020) [12] develop a single model for planning and scheduling, then reschedule the daily appointment times with a secondary model.

Cataldo et al. (2023) [13] develop a heuristic approach for planning and then solve successive models to make daily nurse, chair, and time slot assignments. The first model minimizes the makespan; the second model minimizes the treatment start times under a given makespan.

Another group of studies focus on daily appointment schedules only. Liang and Turkcan (2015) [14] propose models for primary and functional care treatment schemes. In their setting, the nurse is the only resource and has capacity in terms of total acuity levels that s/he can serve. Heshmat et al.(2018) [15] cluster patient-chair pairs using clustering algorithms, then assign clusters to nurses with a mixed-integer programming (MIP) model to minimize the makespan. Hesaraki et al. (2020) [16] assign patients to nurses and time slots such that the first slot covers the set-up.

Benzaid et al. (2020) [17] solve a series of problems, where the first problem chooses patients and sets appointments for a single day and the second problem assigns nurses to the selected patients. Lyon et al. (2023) [18] propose a column-generation method in which each column represents a patient pattern filling time slots. Their primary objective is to minimize the overtime, whereas minimizing the empty slots in a pattern is the secondary objective.

A key difference between our study and the aforementioned studies is that we take the uncertainty of infusion durations into account, which positions our work in chemotherapy scheduling under the stochastic setting. Next, we discuss the pertinent studies and the specific distinctions between each study and ours.

Alvarado and Ntaimo (2018) [19] and Hesaraki et al. (2023) [20] study both planning and appointment scheduling. Since this two-phase problem is hard to manage, they simplify it in different aspects. Alvarado and Ntaimo (2018) [19] formulate a mean-risk stochastic programming (SP) model for a single patient. Unlike Alvarado and Ntaimo (2018) [19], we schedule multiple patients based on a given plan. Hesaraki et al. (2023) [20] develop a heuristic solution, where appointments are forced to fit a fixed template, and test the performance using a simulation model.

There are also a number of studies on scheduling appointments only. Mandelbaum et al. (2020) [21] suggest a data-driven approach for a problem where the chairs are considered as the only resource type. We, however, include nurses in the system and schedule, abiding by an upper limit on their overtime. Castaing et al. (2016) [22] set appointment times under a given sequence served by a single nurse, aiming to minimize the expected total waiting time and makespan. Our study considers multiple nurses and decides the sequence of patients as well as nurse assignments. Hahn-Goldberg et al.(2014) [23] propose a constraint programming approach to schedule appointments under arrival uncertainty.

Slocum et al. (2021) [24] apply a case study in a system where all patients are appointed at the beginning of the day. They propose three appointment slots and assign patients according to their expected infusion duration. Our study introduces an optimization approach that sets an appointment for each patient individually rather than at predetermined times. Slocum et al. (2021) [24]'s schedule is similar to the baseline schedule of the system we work on; hence our study is different. Liang et al. (2015) [25] formulate a deterministic scheduling model to balance the workload of resources, then solve the problem with different objectives and fix different decisions in each phase. Slocum et al. (2021) [24] and Liang et al. (2015) [25] test the performance of the schedule using discrete stochastic event simulation. Yet, the decision processes in both studies ignore the uncertainty.

Gul (2021) [26] carries Castaing et al. (2016) [22]'s study further by adding multiple nurses. In their setting, the sequence is always fixed, and the objective is the expected value of total waiting and nurse overtime. Haghi et al. (2022) [27] model the scheduling problem with two different formulations, where the first is a scheduling formulation and the second is multi-TSP formulation. They use sample average approximation (SAA) scheme to find an approximation for the expected value of the objective function. The expected recourse function is approximated by Monte Carlo simulation. Our study differs from Gul (2021) [26] and Haghi et al. (2022) [27] in two key aspects: resource allocation decisions and model objective. We consider fairness issues and decide on nurse and chair assignments after patients arrive to the OCC (i.e., in the second stage). Our allocation policy aligns the model with actual practice.

Demir et al. (2021) [28] schedule patient appointments by minimizing the weighted averages of patient waiting time, chair idle time, and nurse overtime. Linearized progressive hedging algorithm is proposed to obtain a good-quality solution in reasonable time. Karakaya et al. (2023) [29] is an extension to Demir et al. (2021) [28] and treat nurses as distinct based on their varying skills and patients with their acuity levels; hence, they assign patients to nurses in the first stage. To solve their problem, Karakaya et al. (2023) [29] develop a scenario grouping-based decomposition method that groups scenarios randomly. Gul (2023) [30] investigates how the flexibility in patient-to-nurse assignment policies influences waiting time and overtime. Similar to Karakaya et al. (2023) [29], they develop a decomposition heuristic based on scenario grouping. However, these scenario groups are created through input-based, solution-based, and random grouping methodologies. In terms of the behavior of the nurses, our study is closer to the setting dealt in [28]. The primary distinction in our study from [28], [29] and [30] lies in our approach to the stochastic chemotherapy scheduling problem. We adopt a risk/inequity-averse perspective, introducing a metric that penalizes excess waiting and inequity among patient waiting times. Another significant difference is that we address a nonlinear SP model instead of a linear SP. Lastly, we develop an exact solution method for our model.

2.2 Robust optimization in outpatient appointment scheduling

Two commonly used methods to address uncertainty in optimization problems are stochastic programming and robust optimization. The former is preferred when there is information on the uncertain data while the former is used in the absence of such information. Robust optimization addresses this lack of information by focusing on the extreme cases of the system. As [31] is the only article on chemotherapy scheduling that formulates a robust optimization model, we also review relevant articles from general outpatient appointment scheduling in this subsection.

For general outpatient appointment scheduling problem with uncertain service quality, Behnamian and Gharabaghli (2023) [32] formulate a multi-objective robust optimization model that maximizes the minimum service quality and minimizes the server costs. They propose a multi-objective particle swarm optimization algorithm to solve this problem. Gao et al. (2022) [33] formulate a deterministic model to determine the amount of resources and appointment allocations to these resources. Then, they propose a robust optimization model to minimize resource costs while ensuring that all patients are served even under the worst-case scenario. Shehadeh et al.(2020) [34] studies distributionally robust outpatient colonoscopy scheduling problems. They assume that treatment duration may come from two different distributions based on random preparation quality. They minimize patient waiting time, resource idle time, and overtime. A recent example using a robust programming approach for chemotherapy planning and appointment scheduling is Issabakhsh et al. (2021) [31], which propose a robust slack allocation model and a robust scheduling heuristic to solve small and large-size instances, respectively.

In this study, we address the uncertainty in our problem using a stochastic programming setting. Our approach also considers extreme outcomes by utilizing a metric that controls the waiting time distribution over patients across all scenarios, ensuring that the schedule is fair and efficient even under the worst case, as will be explained in Section 3.

2.3 Fairness in general outpatient scheduling

Fairness has been increasingly considered in the Operations Research area so as to find applicable and acceptable solutions in various real-life systems. Most decision problems arising in public policy, healthcare and humanitarian domains require consideration of fairness concerns for the users and the service providers

[35]. As chemotherapy scheduling problems lie in the area of outpatient scheduling, we discuss studies handling fairness in general outpatient scheduling settings. Fairness, being intrinsic to healthcare provision, should be considered in appointment systems related to healthcare ([36], [37], and [38]). Yet, few papers focus on addressing fairness concerns in appointment scheduling problems, as stated in [6]. Some recent works are [39], [6] and [40].

Turkcan et al. (2011) [39] schedule the appointments in slots considering two unfairness measures under service level constraints. The first is the difference in the number of patients at the beginning of each slot compared to the previous one, and the second is the difference in expected waiting times of patients in each slot. The authors provide a sequential appointment decision algorithm that minimizes the unfairness and maximizes the revenue for a given service level. Unlike this study, we assign the patients simultaneously to multiple servers and do not have fixed slots.

Qi (2017) [6] focuses on a single-phase, single-server setting, and introduces a new metric called Delay Unpleasantness Measure (DUM) based on CVaR. While our metric also draws inspiration from CVaR, it diverges from DUM by focusing on preventing excessive patient waiting times in each scenario rather than preventing excessive scenario outcomes for each patient.

Benjaafar et al. (2023) [40] schedules patients considering server idle time and customer waiting time. They set patient-wise service level constraints to limit the expected waiting time, then optimize the expected idle time. They set a higher utility cost for waiting time above a certain threshold and bring fairness perspective using a patient-wise approach. Compared to this study, we bound the server (nurse) overtime and optimize our proposed metric which limits the extreme outcomes. In terms of solution approaches, while Benjaafar et al. (2023) [40] schedules a single-server system by setting appointment times sequentially, we schedule patients simultaneously for a multi-server system.

The literature review demonstrates that our research stands as the first study in integrating fairness considerations within a chemotherapy scheduling setting.

Chapter 3

Problem Formulation

In this section, we provide an SMINP formulation for the stochastic chemotherapy scheduling problem. Before presenting the model details, we explain the method we propose to ensure equity in waiting times of patients.

3.1 An equity and efficiency encouraging metric

We propose using an *equity and efficiency encouraging metric* (EEEM) that draws inspiration from CVaR, which is a widely used spectral measure. Let W be a random variable describing the outcome of an environment with uncertainty. If smaller values of W are desirable, for a given confidence level α, CVaR is as follows: $\mathrm{CVaR}_\alpha(W) = \min_\eta\{\eta + \frac{1}{1-\alpha}\mathbb{E}([W-\eta]_+)\}$. This is a minimization problem, where the optimality is attained at a specific value of η, called Value at Risk (VaR) and given as $VaR_\alpha(W) = \min\{\eta : F_W(\eta) > \alpha\}$. Equivalently, CVaR can be represented by its LP dual as $\mathrm{CVaR}_\alpha(W) = \frac{1}{1-\alpha}\int_\alpha^1 \mathrm{VaR}_\delta(W)\, d\delta$. According to the last formulation, CVaR indicates the weighted average of the least preferred outcomes in a distribution. In other words, for random variable W with confidence level α; $\mathrm{CVaR}_\alpha(W)$ measures the risk of W as the severity of the realizations worse than $\mathrm{VaR}_\alpha(W)$, which occurs with probability $(1-\alpha)$.

In our setting, minimizing CVaR translates to minimizing excessive waiting times among scenario outcomes for a single patient. However, we are interested in waiting times of multiple patients.

Noyan (2018) [41] discusses multi-variate CVaR case in detail and presents stochastic dominance relations for the setting. A typical aggregate measure for such a setting is the weighted average of CVaR of multiple random variables [42]. In our context, adopting this approach could assist in assessing excessive waiting times among scenario outcomes for individual patients and minimizing the weighted average of those waiting times across all patients. However, it is important to note that this approach may overlook the probability of the co-occurrence of the worst cases in a given scenario. To prevent this, we aim to maintain equity among patients in an arbitrary scenario.

Table 3.1 shows the waiting time values associated with two hypothetical schedules, each designed for four patients and assessed under four scenarios. When the waiting times of each patient are viewed as random variables, the distributions of random variables are clearly identical for a given patient in both schedules. Thus, the CVaR value for a given patient in Schedule 1 is the same as that in Schedule 2. However, if Scenario 1 is realized, Schedule 1 would lead to the worst cases of waiting times for three patients. Yet, the classical approach of minimizing CVaR values treats these two schedules as equivalent.

Table 3.1: Waiting times for “hypothetical example #2” with 4 patients and 4 scenarios

(a) Schedule 1

	Patients			
Scenario	1	2	3	4
1	30	0	35	75
2	15	0	10	0
3	10	0	5	0
4	20	0	15	10

(b) Schedule 2

	Patients			
Scenario	1	2	3	4
1	15	0	15	75
2	30	0	5	0
3	20	0	10	10
4	10	0	35	0

Accordingly, we develop a quantitative measure called EEEM as follows: EEEM $= \inf_{\omega \in \Omega}(\alpha^{\omega})$ which satisfies the following set of inequalities for given bound τ over the set of scenarios Ω.

$$\eta^{\omega} + \frac{1}{|I|(1-\alpha^{\omega})} \sum_{i \in I} [W_i^{\omega} - \eta^{\omega}]_+ \leq \tau, \quad \forall \omega \in \Omega$$

In this approach, EEEM denotes the percentile up to which patients are protected against unacceptable waiting times. In simple terms, our approach differs from the classical CVaR minimization method in the following manner: The traditional approach minimizes the weighted average of $(1-\alpha)$ of the highest waiting times across all scenarios for a specified confidence level α. In contrast, our approach aims to maximize the confidence level α (EEEM) to ensure that the weighted average of waiting times for $(1-\alpha)$ of the longest waiting patients does not exceed a predetermined upper limit in each scenario.

Recall the two example problems illustrated in Tables 1.1 and 3.1; let $\tau = 50$. In the first example (Table 1.1), EEEM values of the schedules are 0.5 and 1, respectively. The second schedule results in fairer waiting time realizations across all scenarios compared to the first one and our metric captures this difference. Similarly in the second example (Table 3.1), EEEM values are 0.375 and 0.571, indicating that our proposed metric can distinguish these outcomes.

Our focus is on achieving high fairness while maintaining a desired level of service performance. The τ parameter in EEEM helps the decision maker to control efficiency of the schedule, i.e. the total waiting time, by enforcing a threshold for the weighted average waiting time across all possible scenarios. Note that this is not possible when one employs inequality measures to incorporate fairness in the schedule design as they quantify the degree of spread, regardless of the total amount allocated.

To demonstrate the difference between EEEM and an inequality measure, we provide four samples in Table 3.2, in which we compare EEEM with the Gini coefficient, which is a well-known inequality measure. Note that the higher the values of EEEM and (1-Gini), the better. We show four possible cases that can be observed depending on the problem structure.

Case 1: A single patient waits for 2 minutes, 1-Gini of the schedule is 0.125, indicating the highest level of unfairness for an 8-patient system. Conversely, EEEM is 1 since all patients wait less than the τ threshold.

Case 2: The total waiting time is again allocated to a single patient; therefore 1-Gini remains 0.125. However, EEEM decreases as it captures the increase in total waiting time. This highlights that as the total waiting time increases, the fairness of the distribution becomes more significant for EEEM.

Case 3: A schedule in which each patient waits for an equal duration is totally fair, hence 1-Gini is 1 in this case. However, this is an undesirable outcome for EEEM. When all of the patients are dissatisfied with the waiting times, equity becomes unimportant.

Case 4: In this case, the total waiting time is 35, as in case 2, but the distribution of waiting times among patients is more equitable. This distinction can be observed in both 1-Gini and EEEM values.

Table 3.2: EEEM with $\tau = 10$ and 1-Gini for "hypothetical example #3"

			patient waiting time							
Schedules	EEEM	1-Gini	1	2	3	4	5	6	7	8
1	1	0.125	0	0	0	0	2	0	0	0
2	0.563	0.125	35	0	0	0	0	0	0	0
3	0	1	50	50	50	50	50	50	50	50
4	0.672	0.255	0	0	8	7	0	0	15	5

In Table 3.3, we present the appointment times for two different schedules and the resulting waiting times under three scenarios. The table shows that Schedule 1 is preferable over Schedule 2 in terms of EEEM, while Schedule 2 is preferable over Schedule 1 in terms of the 1-Gini.

First, let us analyze how these metrics change across scenarios when Schedule 1 is implemented. Scenario 3 demonstrates a superior α^{ω} and $1 - \text{Gini}^{\omega}$ scores than Scenario 2 despite having a longer total waiting time. This is because both metrics consider fairness. In Scenarios 1 and 2, where all waiting times are attributed to a single patient, equity, as indicated by the 1-Gini, is at its lowest. When we compare these scenarios with respect to EEEM, we see that Scenario 2 has lower α^ω, highlighting the influence of efficiency on the metric.

Schedule 2 has a better 1-Gini score than Schedule 1 as the waiting times are distributed more evenly among patients. Note that this metric ignores the increase in total waiting times: the waiting times are drastically higher compared to Schedule 1. EEEM, however, captures this decrease in efficiency. α^ω value is 0 in every scenario, even in Scenario 1, where the total waiting time is distributed equally among most of the patients. Overall, this significant difference between EEEM and Gini comes from their approaches while assessing the schedule. While Gini only checks the distribution of the waiting times over patients, EEEM checks the sum of waiting times as well as their distribution. In a chemotherapy setting, EEEM has the advantage over Gini to create both efficient and fair schedules.

Table 3.3: EEEM with $\tau = 10$ and 1-Gini for "numerical example"

Infusion Durations			Schedule 1	Waiting Times			Schedule 2	Waiting Times		
$\omega = 1$	$\omega = 2$	$\omega = 3$	appointment times	$\omega = 1$	$\omega = 2$	$\omega = 3$	appointment times	$\omega = 1$	$\omega = 2$	$\omega = 3$
120	120	63	15	0	17	0	56	20	37	7
20	39	46	0	0	0	0	0	0	0	0
200	173	141	195	0	0	0	156	20	35	0
180	176	139	0	0	0	0	0	0	0	0
20	63	33	15	0	0	0	15	0	0	0
180	199	174	194	1	0	0	150	20	28	0
126	109	165	54	0	0	7	15	20	39	46
109	98	104	230	0	0	11	175	20	53	66
			$1-\text{Gini}^\omega$	0.125	0.125	0.223	$1-\text{Gini}^\omega$	0.625	0.555	0.21
			α^ω	1	0.788	0.834	α^ω	0	0	0

3.2 Problem description and model formulation

We focus on the problem of sequencing patients and setting appointment times under uncertain infusion durations for a daily list of treatments. We assign patients to nurses and chairs based on an optimal myopic policy. Our goal is to maximize EEEM while ensuring nurses do not exceed overtime restrictions. Since using the EEEM involves nonlinear functions, we propose a two-stage stochastic mixed-integer nonlinear programming (SMINP) model to study the problem.

In the first stage of the model, patient sequencing and appointment time setting decisions are made without perfect information on infusion durations. Our dataset supports the incorporation of uncertainty regarding infusion durations (see Section 5.1 for details). The durations of premedication are assumed to be constant for all patients, as supported by the tight confidence interval estimated based on the same data set. In each realization of infusion duration scenarios, second-stage decisions are made. In this stage, patients are assigned to nurses and chairs, and nurse overtimes, patient waiting times, and EEEM values associated with each scenario are calculated.

Patients are assumed to be punctual, meaning they arrive precisely at their scheduled appointment times ([22, 28, 29]). Consequently, patient waiting time is defined as the time elapsed between the actual start of chemotherapy treatment and the scheduled appointment time for the patient. A patient waits for the treatment until a chair and a nurse become available. When there are multiple patients waiting, the priority is given to the patient who arrives the earliest. In other words, if a patient is scheduled to arrive later than another patient, the chemotherapy treatment of the patient cannot start earlier than that of the preceding patient in the arrival sequence ([28]).

All nurses possess identical skills, and the chairs available in the OCC are also identical. It follows that any patient can be assigned to any available nurse and chair within the clinic. In other words, *functional care delivery* is considered in terms of nursing care delivery scheme ([14]).

At the initial phase of the treatment, patients demand close nursing attention. Therefore, a nurse can serve one patient at a time during the premedication phase. On the other hand, they can monitor multiple patients during the infusion phase. Since their primary task during this phase is to ensure precise drug flow, they can even monitor multiple infusions while managing a patient's premedication. In addition, a nurse can be attending to at most four patients at a time (see also [22] for the same observation). However, we do not explicitly limit the number of patients that can be monitored by a nurse during the infusion phase. The OCC is organized into treatment rooms, with each room accommodating a specific number of chairs and nurses. In the OCC we studied, rooms typically have either three or four chairs, and each room is managed by two nurses who oversee the cases. The ratio of chairs to nurses in a room is carefully managed by OCC management to maintain high-quality treatment. These limits, along with constant premedication durations, ensure that a nurse's workload never exceeds the generally recommended limit of 4 patients at any given time.

Based on the above description, and the notation in Table 3.4, we propose the following SMINP formulation, aiming to create a schedule with the highest EEEM.

Table 3.4: Notation used throughout the two-stage SMINP model

	Index sets
I	Patients
C	Chairs
Ω	Scenarios
N	Nurses
	Parameters
s	Pre-medication time
t_i^ω	Infusion time of patient $i \in I$ in scenario $\omega \in \Omega$
H	Shift duration of nurses
L	Overtime limit of nurses
M	A large value
τ	Waiting time threshold
	First-stage decision variables
EEEM	
b_{ij}	$= \begin{cases} 1, & \text{if patient } i \in I \text{ precedes patient } j \in I \text{ in the daily appointment list} \\ 0, & \text{otherwise} \end{cases}$
a_i	Appointment time of patient $i \in I$
	Second-stage decision variables
x_{in}^ω	$= \begin{cases} 1, & \text{if patient } i \in I \text{ is assigned to nurse } n \in N \text{ in scenario } \omega \in \Omega \\ 0, & \text{otherwise} \end{cases}$
y_{ic}^ω	$= \begin{cases} 1, & \text{if patient } i \in I \text{ is assigned to chair } c \in C \text{ in scenario } \omega \in \Omega \\ 0, & \text{otherwise} \end{cases}$
w_i^ω	Waiting time of patient $i \in I$ in scenario $\omega \in \Omega$
O_n^ω	Overtime of nurse $n \in N$ in scenario $\omega \in \Omega$
q_i^ω	Auxiliary variable for EEEM calculations associated with patient $i \in I$ in scenario $\omega \in \Omega$
η^ω	Value at risk for scenario $\omega \in \Omega$
α^ω	

$$
\begin{aligned}
\max \quad & EEEM && && (3.1)\\
\text{s.t.} \quad & b_{ij} + b_{ji} = 1 && \forall i, j \in I, j > i && (3.2)\\
& a_j \geq a_i - M\big(1 - b_{ij}\big) && \forall i, j \in I, j \neq i && (3.3)\\
& a_i \leq H && \forall i && (3.4)\\
& EEEM \in [0, 1) && && (3.5)\\
& b_{ij} \in \{0, 1\} && \forall i, j \in I, j \neq i && (3.6)\\
& a_i \in \mathbb{Z}^+ && \forall i && (3.7)\\
& (a, b, \text{EEEM}, \xi(\omega)) \in \Xi(\omega) && \forall \omega \in \Omega && (3.8)
\end{aligned}
$$

where $\Xi(\omega)$ represents the feasibility set in the second stage scenario ω, and is described as:

$$\text{s.t.} \sum_{n \in N} x_{in}^{\omega} = 1 \qquad \forall i \in I \tag{3.9}$$

$$\sum_{c \in C} y_{ic}^{\omega} = 1 \qquad \forall i \in I \tag{3.10}$$

$$a_j + w_j^{\omega} \geq a_i + w_i^{\omega} - M\big(1 - b_{ij}\big) \qquad \forall i, j \in I, j \neq i \tag{3.11}$$

$$a_j + w_j^{\omega} \geq a_i + w_i^{\omega} + s\text{-}M\big(3 - b_{ij} - x_{in}^{\omega} - x_{jn}^{\omega}\big) \qquad \forall i, j \in I, j \neq i, \forall n \in N \tag{3.12}$$

$$a_j + w_j^{\omega} \geq a_i + w_i^{\omega} + s + t_i^{\omega} - M\big(3 - b_{ij} - y_{ic}^{\omega} - y_{jc}^{\omega}\big) \ \forall i, j \in I, j \neq i, \forall c \in C \tag{3.13}$$

$$O_n^{\omega} \leq L \qquad \forall n \in N \tag{3.14}$$

$$O_n^{\omega} \geq a_i + w_i^{\omega} + s + t_i^{\omega} - H - M\big(1 - x_{in}^{\omega}\big) \qquad \forall i \in I, n \in N \tag{3.15}$$

$$q_i^{\omega} \geq w_i^{\omega} - \eta^{\omega} \qquad \forall i \in I \tag{3.16}$$

$$\eta^{\omega} + \frac{1}{|I|(1 - \alpha^{\omega})} \sum_{i \in I} q_i^{\omega} \leq \tau \tag{3.17}$$

$$\text{EEEM} \leq \alpha^{\omega} \tag{3.18}$$

$$x_{in}^{\omega} \in \{0, 1\} \qquad \forall i \in I, n \in N \tag{3.19}$$

$$y_{ic}^{\omega} \in \{0, 1\} \qquad \forall i \in I, c \in C \tag{3.20}$$

$$\alpha^{\omega} \in [0, 1) \tag{3.21}$$

$$w_i^{\omega}, q_i^{\omega} \geq 0 \qquad \forall i \in I \tag{3.22}$$

$$\eta^{\omega} \geq 0 \tag{3.23}$$

$$O_n^{\omega} \geq 0 \qquad \forall n \in N \tag{3.24}$$

The objective function (F.1) maximizes the first-stage return which corresponds to the EEEM value that is guaranteed across all scenarios.

Constraint sets (3.2)-(3.7) are the first-stage constraints. Constraints (3.2) ensure that one must precede the other for each pair of patients. Constraint set (3.3) enforces the succeeding patient in the sequence not to arrive earlier than its predecessors. Constraint (3.4) sets the boundary for appointment times and ensures that the patients arrive within the shift hours. Constraint (3.5) represents the bounds of our measure. Constraints (3.6) and (3.7) are binary and integrality restrictions on the first-stage variables, respectively.

Note that our problem is a special type of two-stage stochastic programming problem where the second-stage objective function for given first-stage decision variables and a particular scenario is 0 as our second-stage problem is a feasibility problem.

Constraints (3.9) and (3.10) restrict each patient to be explicitly assigned to one nurse and one chair, respectively. Constraints (3.11) and (3.13) satisfy the scenario-wise precedence relations. For a patient pair $i, j \in S$, if patient i arrives earlier according to patient sequence, (3.11) guarantees the pre-medication start time of patient j is greater than or equal to that of patient i. (3.12) assures that patient $j \in S$ waits for the pre-medication of patient $i \in S$ to finish, in case both are assigned to the same nurse. Similarly, if patients $i, j \in S$ are assigned to the same chair, (3.13) guarantees the infusion of patient i ends before pre-medication of patient j starts.

Constraints (3.14) and (3.15) set upper and lower bounds for overtime values. Upper bound is a predetermined bound given by the OCC manager. The lower bound is determined by the positive difference between the time when the service of a patient assigned to the nurse finishes and the end of the nurse's shift.

Fairness requirements are defined by constraints (3.16) and (3.17). (3.16) calculates the positive part of $w_i^\omega - \eta^\omega$ where η^ω corresponds to VaR of α^ω. q_i^ω taking value 0 for some i, ω indicates that patient i waits less than η^ω in scenario ω, hence they do not contribute to the LHS of constraint (3.17) for that scenario. Positive values of q_i^ω are the excess waiting times, which are to be kept low. Constraints (3.17) calculate the weighted average of the excessive waiting times over patients for each scenario and limits it by the threshold set by the OCC manager. (3.18) defines the objective value, EEEM, as the infimum of α^ω over scenario set Ω. Lastly, constraints (3.19)-(3.24) define non-negativity and binary restrictions on the second-stage variables.

By incorporating the nurse and chair assignment decisions in the second stage, our process aligns with the practical approach where patients are assigned to a chair and a nurse after their arrival at the OCC. This decision is preferred over making assignment decisions while setting appointment times, as it would lead to poorer performance measures given the identical nature of nurses and chairs. In practice, assignment decisions are made dynamically whenever a chair and a nurse become available. Although the decisions are dynamic, our problem does not necessitate a multi-stage SP formulation. The validity of using a two-stage SP formulation is justified by the rule represented by constraint (3.11), which is followed in actual practice. This constraint ensures that the patient arrival sequence remains unchanged when making treatment start decisions, as the urgency levels of arriving patients do not vary. Consequently, whenever the resources become available, the treatment of a waiting patient must commence immediately to enhance schedule efficiency. Therefore, we state the following proposition, whose proof is provided in the online Appendix.

Proposition 1 *Suppose that a feasible second-stage solution exists for a given set of first-stage variable values, and that patient $i \in I$ arrived the earliest among the ready patients. Setting x_{in}^{ω} and y_{ic}^{ω} such that $n \in N$ is the first available nurse and $c \in C$ is the first available chair results in a feasible second-stage solution.*

Proposition 1 highlights that a myopic policy is suitable for chair and nurse assignment decisions, implying that the infusion durations of future patients do not affect resource assignment decisions. Consequently, we independently make those decisions for each scenario at the second stage of the SMINP formulation.

3.3 Model improvement

We use the following cuts so as to reduce solution time of the problem.

Symmetry breaking for chairs: Since the chairs are identical, SMINP has symmetric solutions that only differ with respect to the chair index. In order to break this symmetry, we use an optimal myopic assignment policy as follows: We enforce patient 1 to be assigned to chair 1 (constraint (3.25)). Any patient

$i : 2 \leq i < |C|$ can be assigned to only chair $c : c \leq i$. Therefore, we prevent the assignment of patient i to chairs $i+1$ through $|C|$ (constraint (3.26)).

$$y_{11}^{\omega} = 1 \qquad \forall \omega \in \Omega \quad (3.25)$$

$$y_{ic}^{\omega} = 0 \qquad \forall i \in I, \forall c \in C, 1 < i < c, \forall \omega \in \Omega \quad (3.26)$$

Appointment time fixing: We incorporate optimality cuts by exploiting some properties that an optimal solution must satisfy. Regardless of the difference in infusion durations over scenarios or instances, there are certain conclusions about the appointment times of some patients in the optimal solution of our model, which can be incorporated into the model as optimality cuts.

In a realistic environment, the number of chairs $|C|$ is larger than the number of nurses $|N|$, as a nurse can monitor multiple infusions at a time. This makes it possible to pre-assign the appointment times of the first $|C|$ patients in batches. For example, in a setting with 10 patients, 5 chairs, and 2 nurses: setting the first two patients' appointment times to 0 frees up nurses and chairs early. Recall that the premedication duration is constant and is denoted by s. Hence if the next two patients arrive at time s, 3 chairs would be available, and the nurses would have finished premedications of the previous patients. Similarly, for the fifth patient, the earliest appointment time without waiting would be $2s$. Note that in this example, uncertainty in infusion times prevents pre-assigning an appointment time for the 6th patient. However, in the optimistic scenario, the first patient leaves at $s + t_{min}$, where t_{min} represents the minimum infusion duration over all patients and scenarios. This would be a lower bound for the optimal arrival times of the last five patients. To generalize, let $|C| = k|N| + l$, $k, l \in \mathbb{Z}^+, l < |N|$. We can appoint the first $|C|$ patients to $k+1$ distinct appointment times. Let the set of patients with the same appointment times be $I_m = \{(m-1)|N|+1, (m-1)|N|+2, ..., m|N|\}$ for $m = 1, ..., k$, and $I_{k+1} = \{(k)|N|+1, (k)|N|+2, ..., |C|\}$. Then the appointment times of the patients in I_m are $(m-1)s$, $m = 1, ..., k+1$. Lastly, we enforce the remaining patients to arrive no earlier than time $s+t_{min}$. Despite the potential identification of tighter arrival time limits for patients treated at later time periods, our preliminary analysis indicates that enforcing the same limit for each patient yields similar improvements in solution time as that approach. By defining a new auxiliary binary first-stage variable, θ_{mi}, representing whether patient i belongs to batch m; we represent these cuts through equations (3.27)-(3.40).

$$a_i \leq M\theta_{1i} \qquad \forall i \in I \quad (3.27)$$

$$\sum_{j,j\neq i} b_{ij} - (|I| - |N| - 1) \leq M(1 - \theta_{1i}) \qquad \forall i \in I \quad (3.28)$$

$$a_i - (m-1)s \leq M\theta_{mi} \qquad \forall i \in I \quad (3.29)$$

$$(m-1)s - a_i \leq M\theta_{mi} \qquad \forall i \in I \quad (3.30)$$

$$\sum_{j,j\neq i} b_{ij} - (|I| - (m-1)|N| - 1) \leq M\theta_{mi} \qquad \forall i \in I, m = 2, ..., k+1 \quad (3.31)$$

$$(|I| - (m)|N|) - \sum_{j,j\neq i} b_{ij} \leq M\theta_{mi} \qquad \forall i \in I, m = 2, ..., k \quad (3.32)$$

$$(|I| - |C|) - \sum_{j,j\neq i} b_{ij} \leq M\theta_{k+1,i} \qquad \forall i \in I, \quad (3.33)$$

$$s + t_{min} - a_i \leq M\theta_{k+2,i} \qquad \forall i \in I \quad (3.34)$$

$$(|I| - |C|) - \sum_{j,j\neq i} b_{ij} \leq M(1 - \theta_{k+2,i}) \qquad \forall i \in I \quad (3.35)$$

$$\sum_{m=1,...,k+2} \theta_{mi} = k+1 \qquad \forall i \in I \quad (3.36)$$

$$\sum_{i\in I} \theta_{mi} = |I| - |N| \qquad m = 1, ..., k \quad (3.37)$$

$$\sum_{i\in I} \theta_{k+1,i} = |I| - l \qquad (3.38)$$

$$\sum_{i\in I} \theta_{k+2,i} = |C| \qquad (3.39)$$

$$\theta_{mi} \in \{0, 1\} \qquad m = 1, ..., k+2, \forall i \in I \quad (3.40)$$

Constraint (3.27) fixes the appointment times of the first batch of patients. Similarly, (3.29) and (3.30) set the appointment times of batches 2 through $k+1$. Cuts (3.34) and (3.35) define a lower bound for appointment times of the remaining patients. Lastly, (3.28),(3.31)-(3.32), and (3.36)-(3.38) assign patients to batches based on the patient sequencing decisions.

Symmetry breaking for nurse overtime: Similar to chairs, nurses are identical. Hence, we can add cuts for x_{in}^{ω} variables in a way that is similar to (3.25) and (3.26). Furthermore, we can break symmetric solutions on O_n^{ω} variables by ordering the nurses in descending order of overtime. However, adding any of these two cuts makes the nurses distinct. Thus, they are mutually exclusive; we add (3.41) based on our preliminary analysis.

$$O_n^{\omega} - O_{n+1}^{\omega} \geq 0 \qquad \forall n \in N, n \neq |N|, \forall \omega \in \Omega \quad (3.41)$$

Lower bound for overtime of nurse 1: A nurse works until the latest chair s/he serves is released. Hereby, the highest nurse overtime equals to the latest closure time of chairs minus the regular shift length. The total operation duration of all patients is $\sum_i t_i^{\omega} + |I|s$ for a given scenario. Furthermore, based on the information from the appointment time fixing cut, the total idle time is at least $(\sum_{m=1}^{k} (m-1)|N| + m(|C| - k|N|))s$. Consider an ideal case where the total operation and idle durations are distributed equally over chairs. The closure time in this ideal case is a lower bound to the highest nurse overtime, which is O_1^{ω} due to constraint (3.41). Hence, we obtain constraint (3.42).

$$O_1^{\omega} + H \geq \frac{\sum_i t_i^{\omega} + (|I| + \sum_{m=1}^{k} (m-1)|N| + m(|C| - k|N|))s}{|C|} \qquad \forall \omega \in \Omega \quad (3.42)$$

Chapter 4

Solution Algorithm

Note that the model is a nonlinear SP problem, which makes it hard to solve. One commonly used method to solve SP problems is using decomposition algorithms. Yet, if the recourse type is not complete or relatively complete, then these algorithms may not converge to a solution with a feasible recourse for every scenario. Moreover, although it is possible to solve the model using a non-linear solver, it is not sufficiently practical for this specific problem, as we show in the computational experiments.

Considering the inherent complexity of the problem, we propose a binary search algorithm (BSA) over the possible range of EEEM values, i.e. [0,1) in Section 4.1. This approach relies on the fact that SMINP reduces to a feasibility problem for a given value of EEEM, necessitating to solve it repetitively. However, solving the feasibility problem itself proves to be challenging, even with a moderate number of scenarios. Therefore, we introduce an enhanced version of BSA in Section 4.2.

4.1 Binary search algorithm

Binary search algorithm (BSA) is a search algorithm to find the location of a point. It can locate the target point with a specified precision if the starting search space is continuous, by reducing the search space to half of its previous size at each iteration. Due to this feature of the algorithm, it is a logarithmic search method, and the number of iterations needed is $log_2(\frac{\text{Initial size of the search space}}{\text{Absolute gap for specified precision}})$.

In our problem, the target point EEEM* is defined between [0,1). Hence, we first check whether the problem is feasible for EEEM= 0.5. If it is feasible (resp. infeasible) we update the range as [0.5,1) (resp. [0,0.5)). We iteratively solve these feasibility problems until the precision for the optimal point, i.e. the absolute gap, is less than 0.001. To achieve this precision, we need to check exactly 10 different EEEM values to find EEEM*. The pseudo-code of BSA for our problem is given in the online Appendix.

4.2 Modified binary search algorithm

We modify BSA by implementing numerous procedures to enhance its computational capabilities and propose the *Modified Binary Search Algorithm (MBSA)*. BSA finds the optimal solution of the nonlinear problem by solving a series of linear feasibility models, eliminating the computational burden of solving a nonlinear model. However, the feasibility models are still hard to solve. To address this, we introduce a novel algorithm within MBSA called the “reduce-and-augment” algorithm. This algorithm employs a two-step approach, starting with the reduction of the original scenario set to a smaller size, followed by iterative augmentation until a feasible solution is obtained for the original problem. The first step (scenario reduction) helps in identifying a feasible solution that satisfies the reduced problem’s constraints. To facilitate this process, we put forth various variants of scenario reduction algorithms. However, the ultimate goal is to ensure that the feasible solution of the reduced problem also translates to feasibility for the original problem at hand. Hence, through the iterative augmentation process, we gradually expand the scenario set to find a solution which is also feasible for the original problem.

The MBSA incorporates further enhancements compared to the BSA. As the feasibility problems in different iterations remain largely similar, with the exception of the fixed EEEM value, we leverage this similarity to utilize the information generated at a particular iteration for subsequent iterations. Moreover, instead of testing only the middle point at each iteration as done in the BSA, we selectively test different points based on specific conditions in the MBSA. This conditional selection method (CSM) enhances the exploration of the search space, enabling us to find solutions more effectively. An important distinction in this modified approach is that the number of iterations required to solve an instance is not constant. It improves the solution time of extremely difficult instances by sacrificing from easier ones. For a more comprehensive understanding of these modifications, we provide detailed explanations in subsequent subsections. The pseudocode of the MBSA is provided in Algorithm 1.

4.2.1 Reduce-and-augment algorithm

In this section, we explain the reduce-and-augment algorithm that we use to solve the feasibility problems obtained by fixing EEEM. The algorithm is a two-phase approach that initially reduces the problem by choosing a subset of the scenario set and then augments it by adding scenarios until MBSA can generalize the result of the reduced problem for the original problem.

Reduction is implemented after the Initialization step of the MBSA. Since the procedure is built around the initial reduced problem, the quality of this problem is prominent in the computational efficiency of MBSA. The scenarios selected for the initial reduced problem should be a highly representative subset of the original set of scenarios so that MBSA terminates with minimum augmentation. After all, the augmentation phase aims to increase the representation capability of the reduced problem by extending the reduced scenario set.

In this study, we compare 8 variants of Reduction methods in which the number of scenarios in the reduced problem is given. Two of these are clustering algorithms that choose scenarios step by step. The selection criteria of these methods are choosing the scenario with the furthest distance to the closest scenario in reduced problem set (FDCP) and to the centroid point of the reduced problem set (FDC), respectively. Both rules aim to achieve high dissimilarity within the cluster.

We also implement a Wasserstein distance-based scenario reduction approach, which is introduced in [43]. When infusion duration is a vector of independent random variables, the reduced scenario set and the original scenario set are two probability distributions. Wasserstein metric is a function to find the distance between these two probability distributions, which calculates the probability transferred from one value of the first distribution to another value of the second. When we consider this as a transportation problem between these two distributions, Wasserstein distance corresponds to the minimum total transportation cost. The algorithm aims to create the reduced set that has the minimum Wasserstein distance with the distribution of the original problem. Rujeerapaiboon et al. (2017) [43] has strong results with $l = 1$ and $l = 2$ norms and creates the distribution with both Local search and MIP models, resulting in four variants that we adopt for our algorithm. We refer to these variants in our experiments as MIP W(l_1), MIP W(l_2), LS W(l_1), LS W(l_2), respectively.

In the last two variants of reduction methods, we have incorporated ideas from multi-objective optimization. In multi-objective optimization, representation of the whole non-dominated set by a reduced subset is an important topic as it eases the decision process by reducing the cognitive effort required to choose the most preferred solution. There are well-established methods for measuring the quality of a subset in terms of representatives such as coverage error and diversity [44]. Leveraging these methodologies, we adapt them to address our specific challenge of generating a reduced scenario set which represents the original set of scenarios. We consider two distinct subsets: one aims at minimizing the total coverage distance from the complete set of scenarios, and the other focuses on maximizing the total diversity. We employ integer programming (IP) models to identify these subsets. The first method is referred to as CDM, and the latter as TDM in our experiments.

Pseudo-codes and the mathematical formulations for these variants are in the online Appendix. In calculating distance parameters between scenarios, we use an input based approach and take each scenario as an $|I|$-dimensional point, where the coordinates are the patients' infusion durations. Our computational experiments, the results of which will be provided in the following section, indicate that choosing the scenario with the furthest distance to the closest scenario in the reduced problem set (FDCP) performs best.

In Step 3 of MBSA, the reduced problem is solved. If this problem is infeasible, the original problem is also infeasible. If the reduced problem has a feasible solution, and the corresponding schedule is feasible for all the remaining scenarios in the original scenario set, then it is a solution to the original problem. In these two cases, the reduced problem is sufficiently good at representing the original problem. Hence, if the termination criterion is not met, MBSA should progress to the next iteration and explore alternative values of EEEM. On the other hand, if the feasible schedule for the reduced problem is infeasible in at least one scenario, the reduced problem needs to be augmented by increasing the size of the reduced scenario set to better represent the original problem. In such occurrences, MBSA selects an augmenting scenario among the ones resulting in infeasibility of the reduced problem solution and adds it to the reduced scenario set. We utilize FDCP for selecting a scenario in the augmentation phase due to its superior performance and the advantage of seamlessly incorporating scenarios one-by-one.

4.2.2 Conditional selection method

The results of our preliminary experiments show that not all iterations require the same computational effort. More specifically, feasibility problems that take an infeasible EEEM value tend to need significantly more computational effort. Thus, we can say that, for $\text{EEEM}_1 < \text{EEEM}_2$, testing EEEM_1 is more likely to require less time compared to testing EEEM_2. However, BSA ignores such conditions and divides the search space into two equal pieces. We implement CSM at Step 3 of the MBSA to check the optimality of the current solution, aiming to terminate the algorithm early.

The procedure works as follows. Let at iteration v, the algorithm returns EEEM^v feasible. Then, EEEM^{v+1} is set to $\text{EEEM}^v + \epsilon$ instead of $\frac{\text{EEEM}^v + UB}{2}$. If EEEM^{v+1} is infeasible, then EEEM^v is proven to be optimal within the precision level. In other words, MBSA performs iteration $v + 1$ to validate the solution in such a case. However, if EEEM^{v+1} is also feasible, MBSA finds a better solution without reducing search space significantly. A trade-off exists between terminating the algorithm without facing infeasible problems, which would reduce the computation time, and performing early termination checks that fail to terminate, increasing the computation time. Considering this, MBSA performs the CSM procedure after obtaining a feasible value by taking the middle point of LB and UB only. The procedure is shown in the online Appendix.

4.2.3 Using information from previous models

In MBSA, a series of models are solved subsequently. These models have almost identical formulations with two possible differences; the scenario set and the fixed value for EEEM.

When the augmentation phase is implemented, the resulting model incorporates new constraints specific to the added scenario, while retaining the constraints from the previous model. Since a feasible schedule exists for the former model, this schedule can serve as the initial solution (warm start) for the succeeding model. Likewise, during a new iteration of MBSA, we can utilize schedules generated in previous iterations. Among these schedules, there exists a specific one that is feasible for the current LB under the original scenario set. The only difference is the fixed value of EEEM at constraint (3.18). While this second similarity is relatively minor compared to the first one, we have the flexibility to choose both, only the first one, or none of them. We refer to these options as complete, partial and no warm start in our experiments, respectively.

When MBSA returns to Reduction step after completing an iteration, the algorithm typically begins with a new reduced scenario set in the standard implementation. However, an alternative approach is to utilize the latest augmented reduced scenario set from the previous iteration. Opting for the latter approach carries the drawback of potentially requiring MBSA to solve larger models than necessary. Conversely, employing the former approach may result in redundant repetition of the augmentation steps already performed in the previous iteration.

Algorithm 1 Modified Binary Search Algorithm

Step 1: Initialization
Let $v = 1$
    $LB = 0$, and $UB = 1$
    $\text{EEEM}^v = (LB + UB)/2$
Step 2: Reduction
Generate initial representative scenario group K^v
Step 3: Solve models to update parameters
Solve the reduced problem, RP, based on K^v for EEEM^v
if EEEM^v is infeasible for RP
    Update UB and set EEEM^{v+1} using conditional selection method (CSM)
end if
else
    if EEEM^v is feasible for the remaining scenarios, $\{\Omega \setminus K^v\}$
        Update LB and set EEEM^{v+1} using CSM
    end if
    else
        Step 4: Augmentation
        Update K^v by adding a scenario that causes the solution of the reduced problem to
        become infeasible
        Go to Step 3
    end else
end else
Step 5: Termination check
if $UB - LB < \epsilon$
    return $\text{EEEM}^* = LB$
end if
else
    $v = v + 1$
    Go to Step 2
end else

Chapter 5

Computational Experiments

5.1 Instance generation and data

We conduct our experiments using a set of 10 instances. To generate problem instances, we draw actual infusion durations from a dataset that comprises data on 204 patients collected over 11 days at the OCC of Hacettepe University Oncology Hospital. The number of patients observed in a day ranges between 14 and 23.

The patient data is classified into 4 groups based on planned durations for treatments by consulting with the head nurse. The probabilities of a patient belonging to each specific group are calculated based on the number of observations in each group. The occurrence probabilities, the actual infusion duration interval, mean, and standard deviation for each group are presented in Table 5.1. Furthermore, the distribution of patient groups across days is illustrated in Figure 1(a) in the online appendix. Note that we assume each premedication takes 15 minutes, irrespective of the patient group, as our dataset indicates that 95% confidence interval for premedication durations is in between 14.61 and 16.09.

We generate instances comprising 50 scenarios, where each scenario consists of infusion durations of the patients. To create an instance, we start by sampling a group index for each patient, taking into account the occurrence probabilities of the groups. Subsequently, we sample 50 infusion durations for each patient, considering the duration intervals specific to their associated patient group.

Table 5.1: Occurrence probabilities, infusion duration intervals, means and standard deviations (in minutes) for each patient group

Group	Probability (%)	Duration Interval	Mean	Std. deviation
1	26.96	[16,44]	35.87	11.23
2	7.84	[29,80]	73.31	21.17
3	33.34	[74,132]	126.57	25.62
4	31.86	[125,217]	168.92	25.27

We examine a system comprising 8 patients, 3 chairs, and 2 nurses. We use a tolerance threshold level (τ) of 10 minutes for constraint (3.17). Through our preliminary experiments, we have observed that MBSA can solve instances having optimal values outside the range of (0.375,0.875) substantially fast. Consequently, we adjust the upper bound for nurse overtimes to create a customized instance that enables us to thoroughly test the algorithm's capabilities within a suitable parameter space.

Algorithms are implemented in Microsoft Visual C++ 2022 using CPLEX 20.1 Concert Technology. For the sake of fair comparison, we conduct all experiments on Intel(R) Xeon(R) CPU E5-1650 computer with six-core processor running at 3.60 GHz and 64.0 GB RAM. Additionally, each run has 3-hr time limit to return a result. The solution times are reported in seconds, and the instances terminated by the time limit contribute to the average solution time as 10800 seconds.

5.2 Algorithm experiments

Necessity of modifying BSA: We present compelling numerical evidence that highlights the need for modifying BSA. Our preliminary experiments show that BSA is incapable of finding the optimal EEEM within a reasonable time limit (see Table 5.2). In particular, the termination criterion is not satisfied in 90%

of the test instances within the specified time limit of 3 hours. Furthermore, a substantial gap exists between the latest feasible (proven lower bound) and infeasible (proven upper bound) EEEM values for the majority of the instances. In light of these empirical findings, we investigate and test improvement ideas and implement a modified version, namely the MBSA.

Table 5.2: Relative gap (RG) and absolute gap (AG) between the lower and upper bounds on EEEM values. NS means that the BSA cannot terminate within the time limit.

Instance	Solution Time (sec)	RG(%)	AG	Instance	Solution Time (sec)	RG(%)	AG
1	**NS**	100	1	6	8500.36	1.10	0.001
2	**NS**	25	0.125	7	**NS**	0.25	0.002
3	**NS**	50	0.25	8	**NS**	100	1
4	**NS**	50	0.25	9	**NS**	0.45	0.004
5	**NS**	5.55	0.031	10	**NS**	100	0.5

Comparison of reduction methods: We compare the eight alternative Reduction methods that are discussed in Section 3. Table 5.3 displays the solution times for each instance, as well as the average solution time. The comparison of CDM and TDM methods indicate that the diversity of the scenarios in the reduced set has more impact in solution time than the coverage level of the represented scenarios. A possible explanation can be as follows: the scenarios in a diverse reduced set are also the ones that are hard to be represented by other scenarios.

Moreover, the second-stage problem is a feasibility problem; hence for the reduced set to provide a promising solution for the original problem, not only the total values but also the maximum levels of coverage error and diversity must be considered. Among these eight alternatives, FDCP considers both coverage and diversity; hence outperforms the other approaches with respect to average and maximum solution times. Compared to the second best-performing alternative (TDM), FDCP is 13.88% faster.

Among the Wasserstein distance-based approaches, those utilizing the l_1 norm to measure the distance between two scenarios exhibit notably superior performance compared to those employing the l_2 norm. However, our experiments demonstrate that, overall, the commonly employed Wasserstein distance-based approaches underperform in comparison to the majority of alternative methods on our test instances. This poor performance may be attributed to the unique structure of our second-stage problems. Unlike typical problems addressed by such approaches, our second-stage problems are not conventional optimization problems, but rather feasibility problems.

Table 5.3: Comparison of Reduction methods based on computation times (sec)

Instance	FDCP	FDC	MIP W(l_2)	MIP W(l_1)	LS W(l_2)	LS W(l_1)	CDM	TDM
1	2699.36	7016.31	10008.80	5795.75	3688.80	5854.55	5512.27	7717.08
2	1119.88	1024.25	952.01	386.38	2274.90	1122.79	1318.50	1529.50
3	5196.60	2660.36	10098.60	4764.81	5440.74	4484.21	5617.91	4220.42
4	2490.86	8984.72	1544.79	2267.99	5498.45	3436.52	2962.81	2005.21
5	967.96	1869.78	2466.52	2123.97	2057.16	5094.40	1724.60	1923.95
6	1120.89	1029.55	1226.62	1040.97	1936.99	917.05	9081.99	772.63
7	969.51	492.19	1336.70	1111.98	1355.68	3029.63	721.84	984.80
8	698.04	710.46	**3hr**	2194.33	**3hr**	2298.81	1565.19	891.53
9	601.54	1228.51	1535.66	2930.09	1911.79	1260.73	1224.01	1193.19
10	8820.74	7106.24	5460.57	**3hr**	5365.43	**3hr**	**3hr**	7426.34
Average	2468.54	3212.24	4543.03	3341.63	4032.99	3829.87	4052.91	2866.47

Benefit of conditional selection method: Table 5.4 shows the benefit of using CSM within MBSA. The findings highlight that using CSM and hence benefiting from early termination checks during EEEM updates yield a remarkable improvement in terms of solution time. Importantly, the absence of CSM renders two out of the ten instances unsolvable within the 3-hour time limit, whereas the utilization of CSM ensures successful solution for all instances.

Table 5.4: Comparison of MBSA with CSM against MBSA without CSM in terms of computation times (sec)

With CSM				Without CSM			
Instance	Time	Instance	Time	Instance	Time	Instance	Time
1	2699.36	6	1120.89	1	2277.85	6	801.53
2	1119.88	7	969.51	2	1212.58	7	1847.83
3	5196.60	8	698.04	3	6391.26	8	892.27
4	2490.86	9	601.54	4	**3hr**	9	581.38
5	967.96	10	8820.74	5	3171.86	10	**3hr**
Average	2468.54			Average	3877.65		

Impact of using information from previous models: As outlined in Section 4.2.3, the feasibility problems tackled in the MBSA are similar, hence we can

Table 5.5: Effect of using information from previous models

(a) Effect of complete and partial warm start on solution time (sec)

Instance	Complete	Partial	None
1	2699.36	4105.26	5212.73
2	1119.88	1480.82	3038.62
3	5196.60	3909.09	5156.07
4	2490.86	1872.65	3007.21
5	967.96	1051.84	757.81
6	1120.89	1398.27	850.76
7	969.51	856.12	1471.79
8	698.04	1027.47	1103.08
9	601.53	1366.33	1132.77
10	8820.74	**3hr**	10579.40
Average	2468.54	2786.79	3231.02

(b) Effect of carrying augmented scenario set to next iteration (sec)

Instance	Augmented	Fixed
1	2699.36	5834.57
2	1119.88	2510.77
3	5196.60	9048.64
4	2490.86	5272.39
5	967.96	1395.33
6	1120.89	666.21
7	969.51	1723.72
8	698.04	1496.14
9	601.54	1140.45
10	8820.74	10119.30
Average	2468.54	3920.75

leverage information from previous iterations or models solved within the same iteration. Table 5.5a assesses the benefit of using warm start approaches discussed earlier. We present a comparison between two distinct variants, namely complete warm start and partial warm start, against the case of no warm start. The results underline that information from earlier solutions can be beneficial; as the average solution times with partial and complete warm start strategies are 13.74% and 23.60% better, respectively, compared to no warm start. Based on the results, we choose to employ the complete warm start approach within the MBSA.

As discussed before, two alternative strategies can be employed to establish an initial reduced scenario set when a new iteration of MBSA begins. The first approach involves utilizing the latest augmented scenario set from the previous iteration, and the second approach relies on the initial reduced scenario set of the prior iteration. Each strategy bears its own set of advantages and disadvantages. On one hand, a representative scenario set for a given EEEM level may not be the best-fitting one for a different level. On the other hand, starting from scratch may lead to the risk of redundantly repeating augmentation steps that were already executed in previous iterations. In Table 5.5b, we analyze this trade-off by comparing MBSA, which uses the augmented scenario set, hence the first strategy, with a version which uses the fixed initial reduced scenario set (the second strategy). The first strategy performs better in nine of the ten instances and requires only around 50% of the time needed with the second strategy in most of these. Consequently, we decide to employ the first strategy within the MBSA.

Comparison with a nonlinear programming solver: Since our problem is new in the literature, there are no customized algorithms that we can compare our approach to. Hence, we demonstrate its performance by comparing it to one of the most well-known commercial nonlinear programming solvers (NLPS), Gurobi. We solve the extensive formulation of the SMINP model with improvement cuts, as provided in the online appendix. To ensure a fair comparison, we let the solver to terminate when either 0.001 absolute optimality gap or 3-hr limit is reached, as is the case in MBSA. Table 5.6 presents the solution times, the returned EEEM values and absolute optimality gap reported by Gurobi when the time limit is reached.

Table 5.6: Comparison with an Gurobi

	Solution Time (sec)		**EEEM**		
Instance	MBSA	NLPS	MBSA	NLPS	NLPS Gap
1	2699.36	5194.52	0.71	0.71	0
2	1119.88	7246.08	0.61	0.61	0
3	5196.60	**3hr**	0.74	0.62	0.38
4	2490.86	**3hr**	0.75	0.75	0.25
5	967.96	5560.54	0.56	0.56	0
6	1120.89	**3hr**	0.86	0.86	0.11
7	969.51	1455.70	0.76	0.76	0
8	698.04	3330.36	0.54	0.54	0
9	601.54	**3hr**	0.86	0.86	0.14
10	8820.74	**3hr**	0.74	0.73	0.13

The results show that MBSA is faster than Gurobi in every instance. Moreover, the solver fails to terminate within 3 hours in half of the instances. In three of these (instance # 3,4,9), Gurobi even fails to tighten the initial upper bound on EEEM, which is 1, resulting in notably high absolute gaps.

MBSA always returns the best EEEM value within the given precision since it can solve all instances within the time limit. On the other hand, in half of the instances, Gurobi fails to verify optimality within the time limit and in one instance (instance 3), it returns a remarkably worse solution. The average solution time of MBSA is significantly better than that of the NLPS. Overall, MBSA outperforms the NLPS with a great margin.

5.3 Structure of the optimal (MBSA) schedules

When we analyze the optimal solutions in the experiment set, we observe that the EEEM values of our schedules range from 0.54 to 0.86, due to the variation in patient groups and durations across the instances. To examine how the structure of the optimal sequences varies across instances, we create a heat map showing the average infusion durations of patients assigned to each of the eight positions in the sequences, as presented in Figure 5.1. In other words, we investigate whether long or short infusion duration cases are consistently assigned to specific positions in the optimal sequences.

Interestingly, the first three (i.e., $|C|$) positions generally include patients with very long and very short infusion durations. Moreover, the patient with the longest or shortest average infusion duration is never assigned to the last position. Two primary factors contributing to the latter observation are the shift length and the overtime limit. Assigning the patient with the longest infusion duration to the last position risks exceeding the overtime limit. To avoid this risk, one can set the appointment time of this last patient very early, but this may cause unnecessary waiting times in many scenarios. Similarly, if the patient with the shortest infusion duration is scheduled last, their treatment may need to start during overtime hours in some scenarios due to the long infusion durations of preceding patients. Since a patient's appointment time should be set within regular shift hours, the patient may need to wait excessively until their treatment starts during the overtime period in such cases.

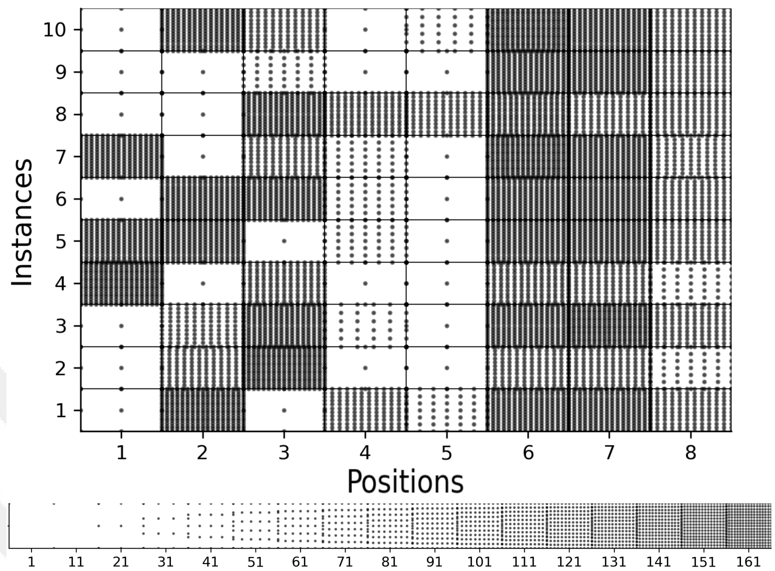


Figure 5.1: Heat map of optimal schedules

5.4 Comparison with alternative schedules

Comparison with baseline schedules: To evaluate the practical efficacy of our SMINP model solution, we conduct a comparative analysis between schedules obtained using MBSA and baseline schedules. The baseline schedules are created by emulating the scheduling approach employed by the head nurse responsible for managing appointments in the OCC that inspired this study.

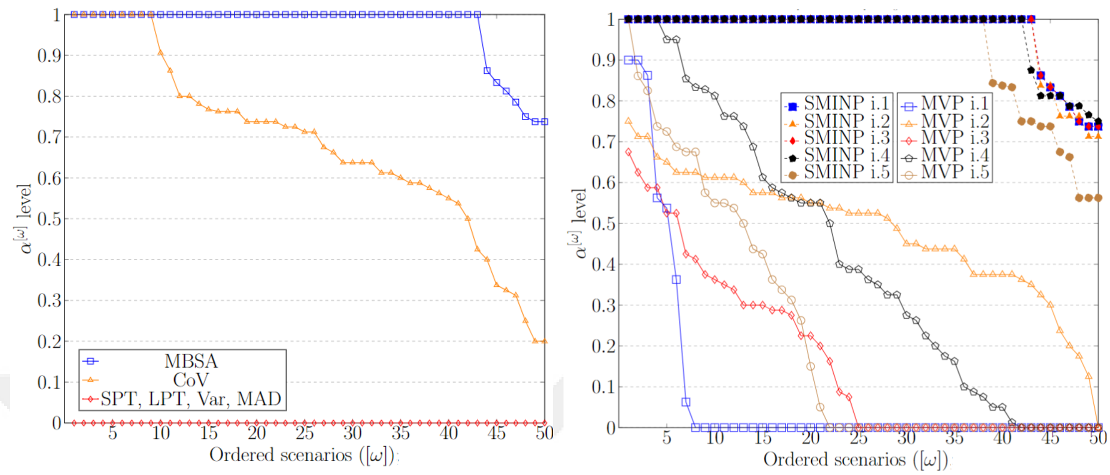
The head nurse assigns patients to two pre-determined appointment times, which are 8:00 a.m. ($a = 0$) and 10:30 a.m. ($a = 150$), respectively. S/he first orders the patients with respect to their expected infusion duration, from minimum to maximum. Starting from the first patient, s/he sets the appointments to 8:00 until the total expected treatment duration per chair exceeds 150 minutes. The remaining appointments are set to 10:30.

For each baseline schedule, we calculate α^ω, $\forall \omega \in \Omega$, through solving a single-scenario subproblem of the SMINP model for scenario ω by fixing sequencing and appointment time variables. Our experiments demonstrate that α^ω is consistently zero for every baseline schedule corresponding to each instance, which indicates that either the average waiting time of scenario ω is greater than τ or the nurse overtime exceeds the limit (both cases make the subproblem infeasible). The result suggests that there is great room for improvement. Our SMINP model fills this need by explicitly considering fairness and efficiency in the resulting schedule.

Comparison with scheduling heuristics: We also compare MBSA schedule with other schedules that can be obtained using some well-known scheduling heuristics. In creating a schedule, we initially employ a sequencing heuristic to order patients and subsequently apply a job hedging heuristic to set the appointment times for all patients [28].

For sequencing decisions, we use the following five rules: 1) shortest infusion duration first (SPT), 2) largest infusion duration first (LPT), 3) least variance in infusion duration first (Var) 4) least coefficient of variation in infusion duration first (CoV), and 5) least mean absolute deviation in infusion duration first (MAD). We take the expected values of the infusion durations while ordering.

We begin by scheduling appointments of the first $|C|$ patients according to the optimal solution property discussed in Section 3.3. Then we estimate the infusion duration of each patient by using job hedging. The sum of constant premedication duration and estimated infusion duration gives the estimated treatment duration for a patient. Next, we add the estimated treatment duration for a patient to the appointment start time of the patient. Subsequently, the appointment time of the next patient is set to the earliest time when both a nurse and a chair become available. In our experiments, we test different job hedging levels between 40^{th} and 100^{th} percentiles, with increments of 5%. Regardless of the sequence and the hedging level, the schedules return EEEM = 0, indicating that such simple heuristics fail to provide good quality solutions. To obtain better schedules, instead of relying on job hedging heuristics to set appointment times, we utilize our SMINP model after establishing the patient sequence through sequencing heuristics. Figure 5.2a shows descending ordered $\alpha^{[\omega]}$ values $\forall \omega \in \Omega$ for a given instance. MBSA has the highest EEEM (i.e., $\alpha^{[50]}$), as expected. However, we see that similar performance can not be obtained by heuristic approaches as the gap between MBSA and the scheduling heuristics solutions is significantly large.



(a) Comparison with scheduling heuristics (b) Value of stochastic solution

Figure 5.2: Scenario-based $\alpha^{[\omega]}$ values of schedules

The waiting time allocation in the worst-case scenario of MBSA is better than or equal to the allocations in 32 scenarios of the best-performing sequencing heuristic (CoV).

Comparison with the min-max approach: We compare our schedules with those of the min-max approach, which aims to prevent the occurrence of extremely unfavorable outcomes. To accomplish this, we solve a revised version of our problem where we minimize the maximum patient waiting time ($w_{\max}$). Solving the min-max problem is computationally demanding. To expedite the solution process, we input the MBSA schedules as warm start solutions. Notably, even with warm start, solving the min-max model requires more than 3 hours in five instances.

In four instances, both schedules (MBSA and min-max) yield the same maximum waiting times, with identical schedules observed in two of them. Over 10 instances, MBSA schedules have 14% higher maximum waiting times whereas min-max schedules have 17% lower EEEM, on average.

Min-max approach prioritizes the worst outcome, which may lead to high inefficiency. We assess the efficiency of these schedules by comparing the total patient waiting times. We observe that the average total waiting time is 9.8%

higher for the min-max schedules. In worst-case scenarios, which are the ones both approaches optimize over, the min-max approach resulted in a 31.43% higher total waiting time on average, reaching up to 80%.

We analyze the two approaches in more detail in Table 5.7. The table displays details of the optimal solutions of the $\min w_{\max}$ and max EEEM approaches, respectively, in instance 1. For these solutions, we provide details on the waiting time distributions for a set of scenarios. In the upper part of the table, details on the optimal solution with respect to $\min w_{\max}$ is seen. This solution has a maximum waiting time of 16, which is realized in six scenarios. Moreover, when we calculate α^{ω} values corresponding to this solution, we see that the worst value is 0.6, lower than the EEEM of the max EEEM (MBSA) solution, which is 0.713. We also show the other scenarios that result in a α^{ω} that is lower than this EEEM. In the second (lower) part of the Table, we provide details on the max EEEM (MBSA) solution, which has an EEEM value of 0.713, observed in two of the scenarios. This solution results in a higher maximum waiting time than 16 in four scenarios, the details of which are given. In these scenarios the maximum waiting time is 19.

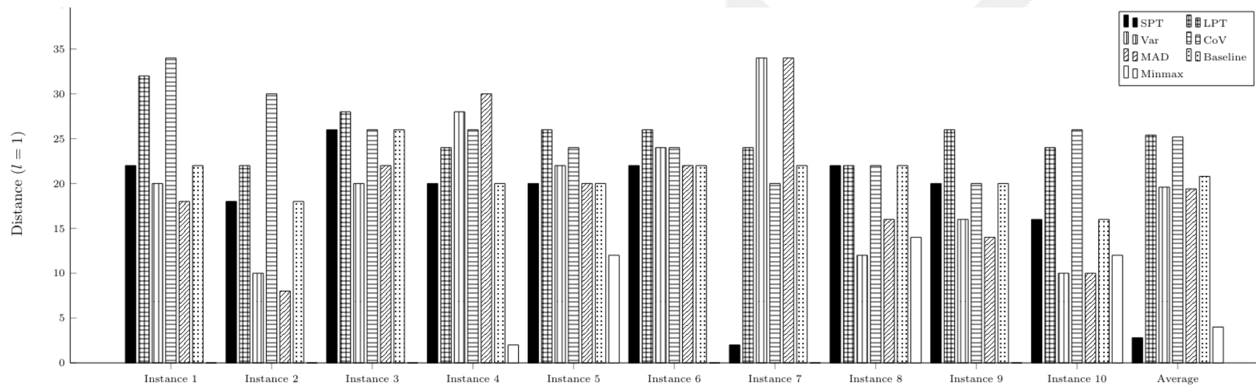
The table reveals a higher likelihood of encountering undesirable scenarios in a min-max waiting time schedule. Additionally, we observe a scenario in which the min-max schedule keeps $w_{\max}$ at its minimum (16) but makes other patients wait, too (see patients 6 and 5, who wait for 11 and 5 minutes, respectively), sacrificing from the overall welfare. This supports our observation regarding the total waiting time difference between the two approaches. We see that the total waiting time can be as high as 32 in the $\min w_{\max}$ solution while it is at most 23 in the max EEEM (MBSA) solution.

Table 5.7: Comparison with min $w_{\max}$ objective

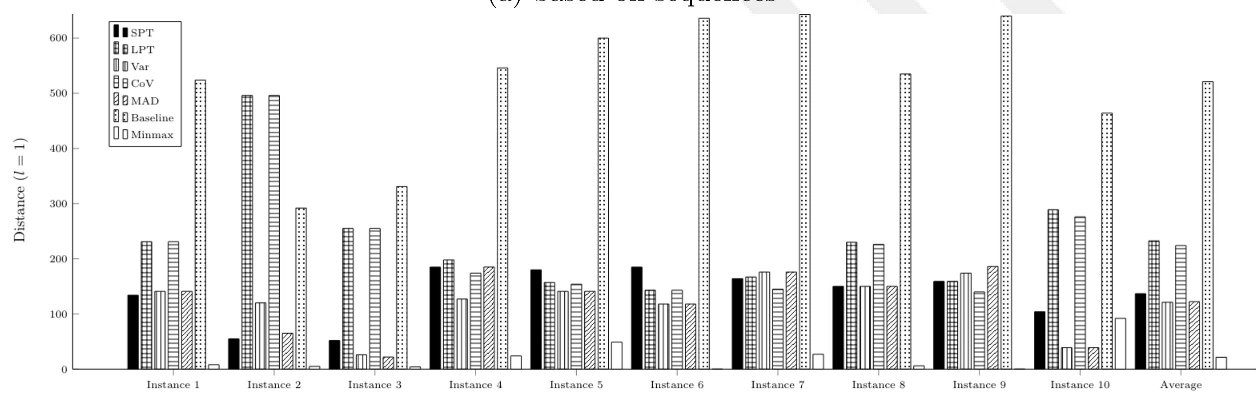
Objective	**Patient**	**Scenarios with $w^{\omega}_{\max} \geq w^{*}_{\max}$**						**Scenarios with $\alpha^{\omega} \leq$ EEEM***			
min $w_{\max}$	1	0	0	0	0	0	0		0	0	0
	2	0	0	0	0	0	0		0	0	0
	3	0	0	0	0	0	0		0	0	0
	4	16	0	0	0	0	16		16	0	0
	5	7	0	16	0	5	0		7	16	5
	6	0	0	0	0	11	0		0	0	11
	7	2	0	9	0	16	0		2	9	16
	8	0	16	0	16	0	0		0	0	0
	$w^{\omega}_{\max}$	**16**	**16**	**16**	**16**	**16**	**16**	α^{ω}	**0.688**	**0.688**	**0.6**
max EEEM (MBSA)	1	0	0	0	0				0	0	
	2	0	0	0	0				0	0	
	3	0	0	0	0				0	0	
	4	19	0	0	19				0	19	
	5	0	0	0	4				0	4	
	6	0	0	0	0				9	0	
	7	0	0	0	0				14	0	
	8	0	19	19	0				0	0	
	$w^{\omega}_{\max}$	**19**	**19**	**19**	**19**			α^{ω}	**0.713**	**0.713**	

$w^{*}_{\max}$ is the optimal value of the min $w_{\max}$ problem,
EEEM* is the optimal value of the max EEEM problem.

Dissimilarity of schedules: In addition to comparing the MBSA schedules to alternative schedules based on performance metrics, we analyze their (dis)similarity. We consider each schedule in terms of its two key components: appointment times and sequences. We then quantify the similarity of a schedule to the MBSA schedule using the $l = 1$ norm. Distances are measured by treating appointments and patient positions as points in a $|I|$-dimensional space, as illustrated in Figure 5.3 (see also the online Appendix for detailed results). Figure 5.3a demonstrates that heuristic approaches provide very different sequences compared to MBSA, whereas min-max sequences exhibit a notable similarity to MBSA sequences. Figure 5.3b shows a similar trend in appointment times. Note that baseline schedules constitute an exception: they exhibit significantly large appointment time distances despite not necessarily having the largest sequence distances. This occurs because of the appointment slot structure utilized in the OCC. The head nurse has to assign patients to only two pre-determined appointment times within the shift, leading to significant deviations from optimal schedules.



(a) based on sequences



(b) based on appointment times

Figure 5.3: Dissimilarity of alternative schedules with MBSA schedule

5.5 Value of stochastic solution

Value of stochastic solution (VSS) is important to assess the impact of uncertain factors on the first-stage decisions. To calculate VSS, we first generate a schedule with the mean value problem (MVP), which has a deterministic formulation obtained by taking the expected value of infusion durations. Subsequently, we compare the objective function value of SMINP solution with that of the mean value problem solution. The outcomes of this experiment show that the deterministic schedule consistently results in EEEM value of 0 for all instances. Figure 5.2b shows descending ordered $\alpha^{[\omega]}$ values $\forall\omega \in \Omega$ of optimal schedules generated by MVP and SMINP formulations for five instances. It is seen that in MVP not only EEEM is worse but also there is a significant gap in $\alpha^{[\omega]}$ values between the proposed and the MVP solutions. Consequently, the VSS value reaches 100%, highlighting a significant disparity in patient waiting times in outpatient chemotherapy schedules when uncertainty in infusion durations is not considered while setting appointment times for patients.

Chapter 6

Conclusion

In this study, our primary objective is to tackle the issue of excessive waiting times in chemotherapy appointments, ensuring fair scheduling for all patients. To achieve this goal, we introduce an equity and efficiency encouraging metric (EEEM) and seamlessly incorporate it into an SMINP model. Utilizing the SMINP model, we sequence patients and set appointment times while accounting for the uncertainties in infusion durations for each patient on a daily list. In our pursuit of equitable scheduling, we also assign patients to nurses and chairs following an optimal myopic policy. Addressing the complexity of nonlinear constraints within the SMINP model, we present a modified version of the binary search algorithm, MBSA. This sophisticated algorithm requires solving stochastic feasibility problems at each iteration. To overcome this challenge, we devise a novel reduce-and-augment algorithm. Multiple variants of scenario reduction algorithms are extensively tested as part of the reduce-and-augment process. Additional improvement procedures such as conditional selection method and warm start techniques are also incorporated to the MBSA. To evaluate the efficacy of the MBSA, we conduct a comprehensive comparison against a commercial nonlinear programming solver (NLPS) using actual data from an OCC of a major oncology hospital. Furthermore, we conduct a comparison of the MBSA solutions with those of the baseline schedules. Finally, we measure the value of stochastic solution.

We found that our exact algorithm, MBSA, outperforms the NLPS significantly in terms of the computational time. Furthermore, the MBSA yields schedules that are remarkably better than both baseline schedules and those generated by commonly used scheduling heuristics. Scenario set reduction and augmentation methods play an important role within the MBSA algorithm, and FDCP (selecting the scenario that is furthest away from the closest scenario in the reduced set) performs the best among several other variants. The VSS calculation underscores the utmost importance of incorporating uncertainty in infusion durations when establishing patient appointment times to maintain fairness in outpatient chemotherapy schedules. By accounting for these uncertainties, we can significantly improve the equitable distribution of waiting times for patients, leading to more effective and patient-centric scheduling strategies.

Our approach is generalizable to various OR/IE applications, with some problem specific modifications. Hence, the potential impact is not limited to chemotherapy scheduling and can be seen in a wide range of applications involving fairness concerns in a stochastic system, including but not limited to general scheduling, workload allocation, humanitarian logistics, hazmat transportation, equitable location, healthcare provision, and public service design. The promising results we obtain for the chemotherapy application indicate that it has potential to improve such systems, too.

In a future study, EEEM measure can be modified and defined for each scenario independently. The objective function of the SMINP would maximize the summation of the k worst EEEM values in such a case. In an alternative extension of the study, a two-level approach can be formulated to favor highly preferred and avoid unfavorable outcomes at the same time. EEEM metric can also be used in problems from settings that are different from chemotherapy clinics to assess its general practicability.

Bibliography

[1] American Cancer Society, "Cancer facts & figures 2024," 2024.

[2] H. Katayama, M. Tabata, T. Kubo, K. Kiura, J. Matsuoka, and Y. Maeda, "Demand for weekend oupatient chemotherapy among patients with cancer in japan," *Supportive Care in Cancer*, vol. 29, pp. 1287–1291, 2021.

[3] F. Camacho, R. Anderson, A. Safrit, A. S. Jones, and P. Hoffmann, "The relationship between patient's perceived waiting time and office-based practice satisfaction," *North Carolina Medical Journal*, vol. 67, no. 6, pp. 409–413, 2006.

[4] O. Baron, O. Berman, D. Krass, and J. Wang, "Strategic idleness and dynamic scheduling in an open-shop service network: Case study and analysis," *Manufacturing & Service Operations Management*, vol. 19, no. 1, pp. 52–71, 2017.

[5] K. McCarthy, H. M. McGee, and C. A. O'Boyle, "Outpatient clinic waiting times and non-attendance as indicators of quality," *Psychology, Health & Medicine*, vol. 5, no. 3, pp. 287–293, 2000.

[6] J. Qi, "Mitigating delays and unfairness in appointment systems," *Management Science*, vol. 63, no. 2, pp. 566–583, 2017.

[7] X. Huang, "Patient attitude towards waiting in an outpatient clinic and its applications," *Health Services Management Research*, vol. 7, no. 1, pp. 2–8, 1994.

[8] P. K. Lau, M. J. Watson, and A. Hasani, "Patients prefer chemotherapy on the same day as their medical oncology outpatient appointment," *Journal of Oncology Practice*, vol. 10, no. 6, pp. e380–e384, 2014.

[9] G. Lamè, O. Jouini, and J. S.-L. Cardinal, "Outpatient chemotherapy planning: A literature review with insights from a case study," *IIE Transactions on Healthcare Systems Engineering*, vol. 6, no. 3, pp. 127–139, 2016.

[10] M. Hadid, A. Elomri, T. E. Mekkawy, O. Jouini, L. Kerbache, and A. Hamad, "Operations management of outpatient chemotherapy process: An optimization-oriented comprehensive review," *Operations Research Perspectives*, vol. 9, p. 100214, 2022.

[11] A. Turkcan, B. Zeng, and M. Lawley, "Chemotherapy operations planning and scheduling," *IIE Transactions on Healthcare Systems Engineering*, vol. 2, no. 1, pp. 31–49, 2012.

[12] P. Hooshangi-Tabrizi, I. Contreras, N. Bhuiyan, and G. Batist, "Improving patient-care services at an oncology clinic using a flexible and adaptive scheduling procedure," *Expert Systems with Applications*, vol. 150, p. 113267, 02 2020.

[13] A. Cataldo, S. Sufan, A. Lorca, M. Andresen, C. Sanchez, and A. Saure, "Multi–stage process for chemotherapy scheduling and effective capacity determination," *International Transactions in Operational Research*, vol. 30, no. 1, pp. 151–180, 2023.

[14] B. Liang and A. Turkcan, "Acuity-based nurse assignment and patient scheduling in oncology clinics," *Health care management science*, vol. 19, 01 2015.

[15] M. Heshmat, K. Nakata, and A. Eltawil, "Solving the patient appointment scheduling problem in outpatient chemotherapy clinics using clustering and mathematical programming," *Computers & Industrial Engineering*, vol. 124, 07 2018.

[16] A. Hesaraki, N. Dellaert, and T. de Kok, "Integrating nurse assignment in outpatient chemotherapy appointment scheduling," *OR Spectrum*, vol. 42, pp. 1–29, 12 2020.

[17] M. Benzaid, N. Lahrichi, and L.-M. Rousseau, "Chemotherapy appointment scheduling and daily outpatient–nurse assignment," *Health Care Management Science*, vol. 23, 03 2020.

[18] G. Lyon, A. Cataldo, G. Angulo, P. A. Rey, and A. Saure, "A column generation approach to intraday scheduling of chemotherapy patients," *International Journal of Production Research*, vol. 61, no. 7, pp. 2231–2249, 2023.

[19] M. Alvarado and L. Ntaimo, "Chemotherapy appointment scheduling under uncertainty using mean-risk stochastic integer programming," *Health Care Management Science*, vol. 21, no. 1, p. 87 – 104, 2018.

[20] A. Hesaraki, N. Dellaert, and T. de Kok, "Online scheduling using a fixed template: the case of outpatient chemotherapy drug administration," *Health Care Management Science*, vol. 26, pp. 117–137, 3 2023.

[21] A. Mandelbaum, P. Momčilović, N. Trichakis, S. Kadish, R. Leib, and C. A. Bunnell, "Data-driven appointment-scheduling under uncertainty: The case of an infusion unit in a cancer center," *Management Science*, vol. 66, no. 1, pp. 243–270, 2020.

[22] J. Castaing, A. Cohn, B. T. Denton, and A. Weizer, "A stochastic programming approach to reduce patient wait times and overtime in an outpatient infusion center," *IIE Transactions on Healthcare Systems Engineering*, vol. 6, no. 3, pp. 111–125, 2016.

[23] S. Hahn-Goldberg, J. C. Beck, M. W. Carter, M. Trudeau, P. Sousa, and K. Beattie, "Solving the chemotherapy outpatient scheduling problem with constraint programming," *Journal of Applied Operational Research*, vol. 6, p. 135–144, 5 2014.

[24] R. F. Slocum, H. L. Jones, M. T. Fletcher, B. M. McConnell, T. J. Hodgson, J. Taheri, and J. R. Wilson, "Improving chemotherapy infusion operations

through the simulation of scheduling heuristics: a case study," *Health Systems*, vol. 10, no. 3, pp. 163–178, 2021.

[25] B. Liang, A. Turkcan, M. E. Ceyhan, and K. Stuart, "Improvement of chemotherapy patient flow and scheduling in an outpatient oncology clinic," *International Journal of Production Research*, vol. 53, no. 24, pp. 7177–7190, 2015.

[26] S. Gul, "Chemotherapy appointment scheduling under uncertainty by considering workload balance among nurses," *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi*, vol. 27, no. 4, pp. 570 – 578, 2021.

[27] M. Haghi, H. Hashemi Doulabi, I. Contreras, and N. Bhuiyan, "Integrated consultation and chemotherapy scheduling with stochastic treatment times," *Journal of the Operational Research Society*, 2022.

[28] N. B. Demir, S. Gul, and M. Çelik, "A stochastic programming approach for chemotherapy appointment scheduling," *Naval Research Logistics*, vol. 68, no. 1, p. 112 – 133, 2021.

[29] S. Karakaya, S. Gul, and M. Çelik, "Stochastic scheduling of chemotherapy appointments considering patient acuity levels," *European Journal of Operational Research*, vol. 305, no. 2, pp. 902–916, 2023.

[30] S. Gul, "Nursing care flexibility in chemotherapy appointment scheduling," *Flexible Services and Manufacturing Journal*, vol. in press, 2023.

[31] M. Issabakhsh, S. Lee, and H. Kang, "Scheduling patient appointment in an infusion center: a mixed integer robust optimization approach," *Health Care Management Science*, vol. 24, pp. 117–139, 2021.

[32] J. Behnamian and Z. Gharabaghli, "Multi-objective outpatient scheduling in health centers considering resource constraints and service quality: a robust optimization approach," *Journal of Combinatorial Optimization*, vol. 45, no. 2, 2023.

[33] Y. Gao, Q. Zhang, C. K. Lau, and B. Ram, "Robust appointment scheduling in healthcare," *Mathematics*, vol. 10, no. 22, 2022.

[34] K. S. Shehadeh, A. E. Cohn, and R. Jiang, "A distributionally robust optimization approach for outpatient colonoscopy scheduling," *European Journal of Operational Research*, vol. 283, no. 2, pp. 549–561, 2020.

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

[36] T. Cayirli and E. Veral, "Outpatient scheduling in health care: a review of literature," *Production and Operations Management*, vol. 12, no. 4, pp. 519–549, 2003.

[37] D. Gupta and B. Denton, "Appointment scheduling in health care: Challenges and opportunities," *IIE Transactions*, vol. 40, no. 9, pp. 800–819, 2008.

[38] A. Ahmadi-Javid, Z. Jalali, and K. J. Klassen, "Outpatient appointment systems in healthcare: A review of optimization studies," *European Journal of Operational Research*, vol. 258, no. 1, pp. 3–34, 2017.

[39] A. Turkcan, B. Zeng, K. Muthuraman, and M. Lawley, "Sequential clinical scheduling with service criteria," *European Journal of Operational Research*, vol. 214, no. 3, pp. 780–795, 2011.

[40] S. Benjaafar, D. Chen, R. Wang, and Z. Yan, "Appointment scheduling under a service-level constraint," *Manufacturing & Service Operations Management*, vol. 25, no. 1, pp. 70–87, 2023.

[41] N. Noyan, *Risk-Averse Stochastic Modeling and Optimization*, pp. 221–254. 10 2018.

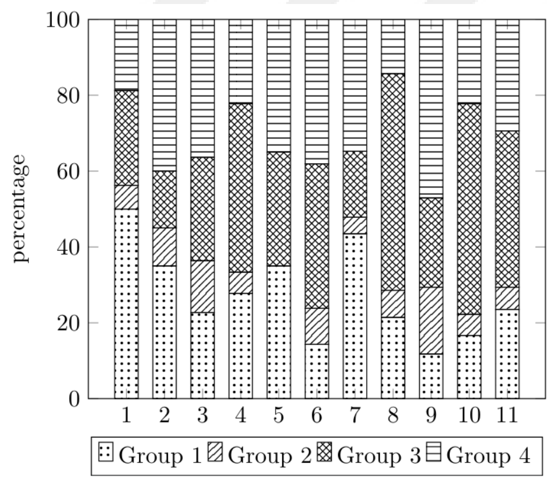
[42] R. Rockafellar, "Solving stochastic programming problems with risk measures by progressive hedging," *Set-Valued and Variational Analysis*, vol. 26, 12 2018.

[43] N. Rujeerapaiboon, K. Schindler, D. Kuhn, and W. Wiesemann, "Scenario reduction revisited: Fundamental limits and guarantees," *Mathematical Programming*, vol. 191, 01 2017.

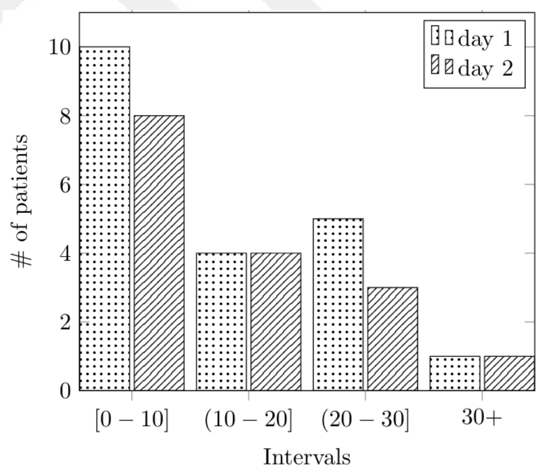
[44] S. Sayın, "Measuring the quality of discrete representations of efficient sets in multiple objective mathematical programming," *Mathematical Programming*, vol. 87, pp. 543–560, 2000.

Appendix A

Details on Data



(a) Distribution of patient groups over 11 days



(b) # of patients waiting within time ranges observed on randomly selected two days

Figure A.1: Observations on data set

Appendix B

Proof of Proposition 1

Let $\mathbf{A} = \{(a_i, b_{ij}), i, j \in I\}$ be the appointment and precedence set from the first stage; and for given $\mathbf{A}$, let S_{FAR} (which will be referred as schedule) be the set of patient-chair and patient-nurse assignments determined by assigning patients to the first available resources. Recall that if $b_{ij} = 1$, patient j must wait for patient i's service to start. We will prove the optimality of S_{FAR} by showing that any optimal schedule S_{opt} can be converted to S_{FAR} by a set of moves without changing the waiting time of any patient.

In an arbitrary step during transforming S_{opt} into S_{FAR}, let S_{inter} be the intermediate schedule obtained at the previous step and $i \in I$ be the earliest arriving patient in S_{inter} among the patients who are assigned to a different nurse and/or chair compared to S_{FAR}.

Let C_1 and N_1 denote the chair and nurse that patient i is assigned under S_{inter} respectively, similarly C_2 and N_2 are equivalent to their counterparts under S_{FAR}.

We can investigate this step under 3 distinct cases.

Case 1: $C_1 \neq C_2$, $N_1 = N_2$

Figure B.1 illustrates this case. Let $j \in I$ be the first patient assigned to C_2 after the service of i starts on C_1, and let k be the last patient assigned to C_2 immediately preceding j. In S_{inter}, we represent the subsets of assigned patients on C_1 as before and after patient i with Subsequence C_{1pre} and C_{1suc} respectively. Moreover, let Subsequence C_{2pre} denote the subset of assignments on C_2 that precede k and Subsequence C_{2suc} those succeeding j. Note that, as $C_2 \in S_{FAR}$, C_2 must be the first available chair for patient i. Thus the service of k finishes before the service of i starts on C_1. Additionally, the model ensures that patient j waits for i's service start time. Figure B.1(a) shows S_{inter} as described.

Next, by retaining the service start timings and nurse assignments of each patient precisely, we can exchange the chair assignment of the subsequence containing patient i and C_{1suc} with the subsequence containing patient k and C_{2suc} on chairs C_1 and C_2. Figure B.1(b) shows the schedule obtained after this exchange. In this case, each patient's waiting time remains unchanged (as their service start times are identical in both schedules); thus, each nurse's overtime is less than the given upper limit ($N_1 = N_2$). Combining these results implies that as S_{opt} is feasible under given optimal EEEM value (EEEM*), S_{FAR} is also feasible.

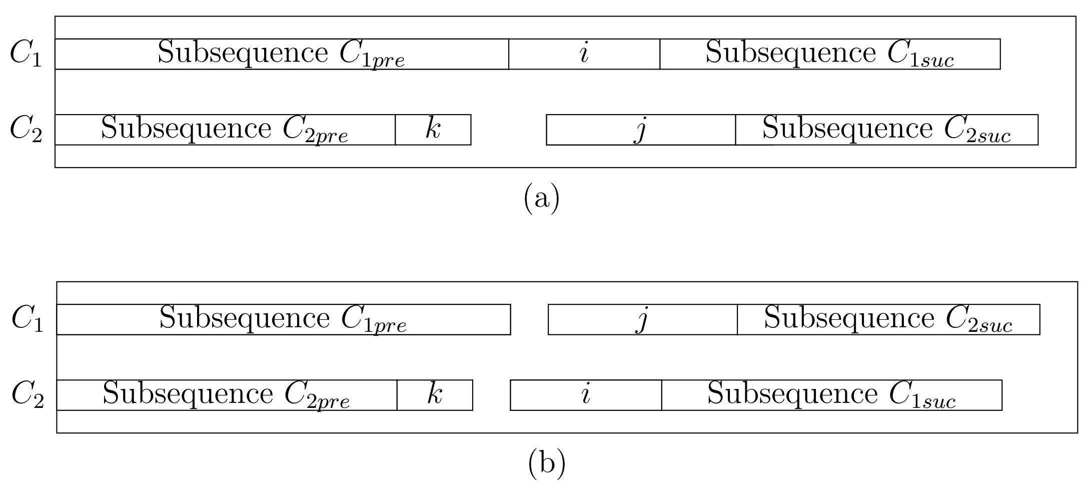
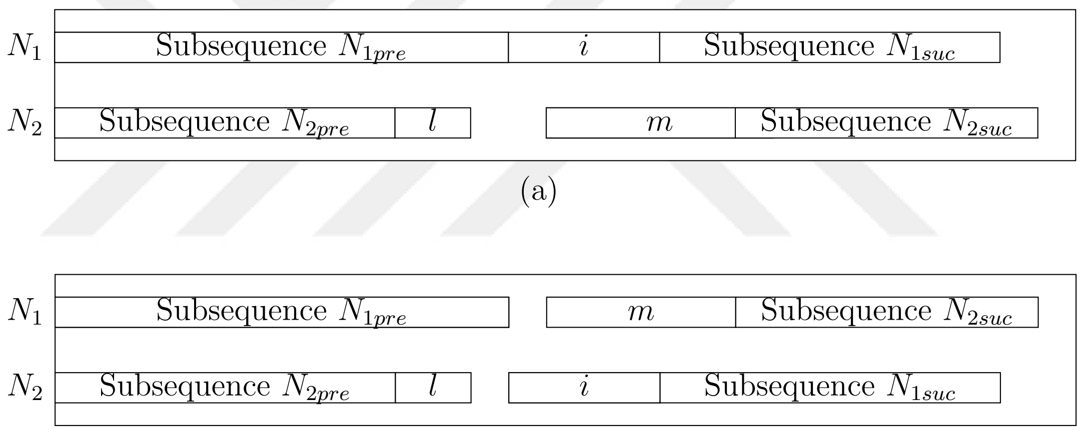


Figure B.1: The patient assignments for chairs C_1 and C_2 (a) under S_{inter} and (b) after exchanging the subsequences under Case 1 ($C_1 \neq C_2$, $N_1 = N_2$)

Case 2: $N_1 \neq N_2$, $C_1 = C_2$

Both nurses and chairs work similarly as a resource. Hence we can apply similar exchange moves as in Case 1. Figure B.2 (a) shows S_{inter} where m represents the first patient assigned to nurse N_2 arriving after patient i, and l is the latest predecessor of m assigned to the same nurse. While keeping the chair assignment, we can swap the subsequences starting from i and m on N_1 and N_2, respectively. The new schedule (see Figure B.2 (b)) has the same service start time for each patient. This implies that the exchange does not break the feasiblity under given EEEM*.



(b)

Figure B.2: The patient assignments for nurses N_1 and N_2 (a) under S_{inter} and (b) after exchanging the subsequences under Case 2 ($N_1 \neq N_2$, $C_1 = C_2$)

Case 3: $N_1 \neq N_2$, $C_1 \neq C_2$

In this case, we can prove the equivalence of the schedules by exchanging the subsequences based on the principles of Case 1 and Case 2 sequentially. Figure B.3 exemplifies the transition. According to results from Case1 and Case 2, the schedule obtained here is feasible under given EEEM*.

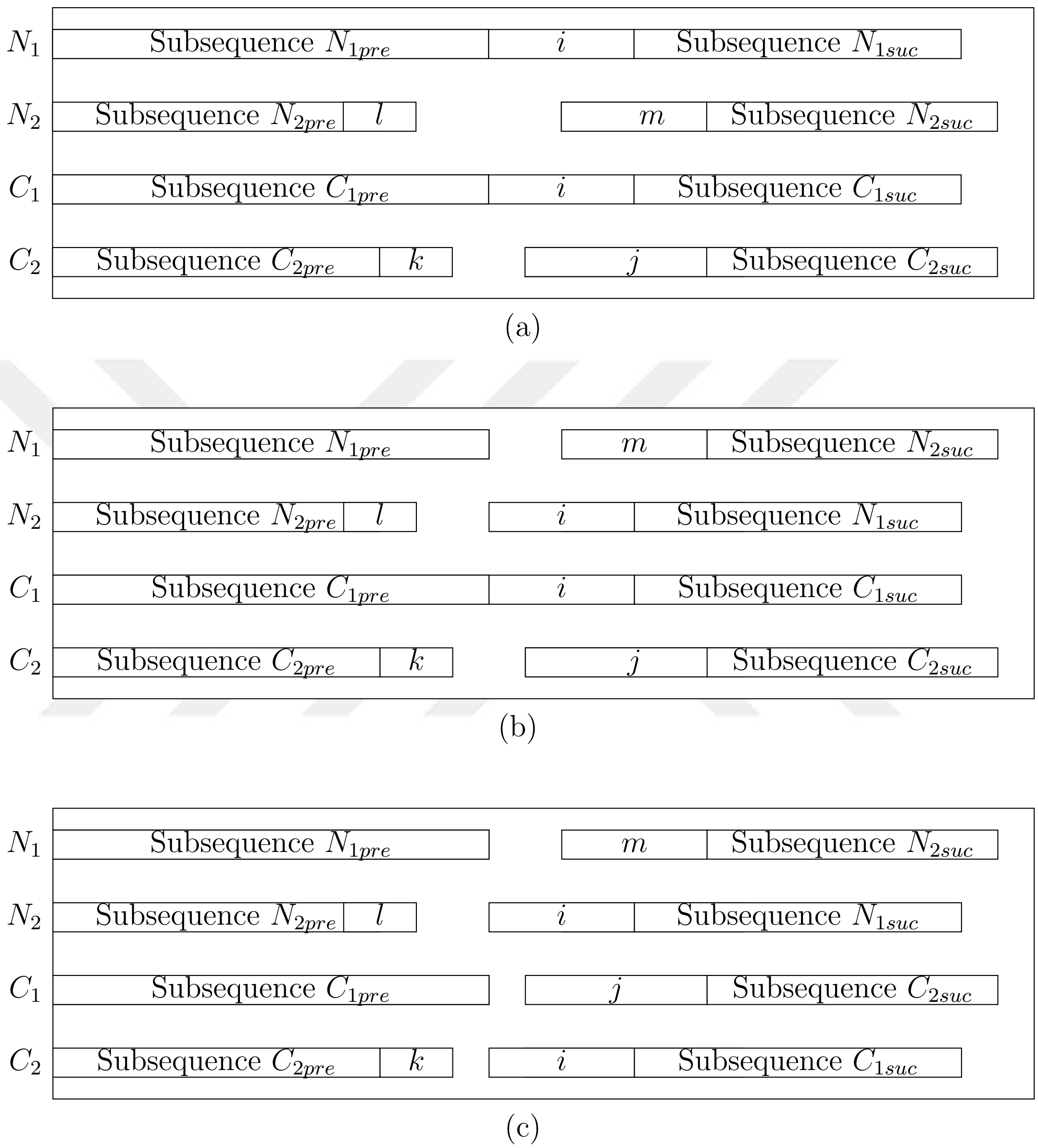


Figure B.3: The patient assignments for chairs C_1 and C_2 and nurses N_1 and N_2 (a) under S_{inter} and (b) after exchanging the subsequences on nurses (c) after exchanging the subsequences on nurses and then on chairs under Case 3 ($N_1 \neq N_2$, $C_1 \neq C_2$)

One can convert S_{opt} to S_{FAR} as follows: Starting from S_{opt}, pick the earliest arriving patient. If the patient is assigned to a different (i) chair, (ii) nurse, or (iii) both compared to those in S_{FAR}, perform the exchange steps described in Case (1), Case (2), or Case (3) respectively. As the schedules are equivalent, one can finish this conversion with at most $|I|$ iterations.

Appendix C

Binary Search Algorithm

Algorithm 2 Binary Search Algorithm

Step 1: Initialization
Let $v = 1$
$LB = 0$ and $UB = 1$
$\text{EEEM}^v = (LB + UB)/2$
Step 2: Solve the feasibility problem for EEEM^v
Step 3: Update algorithm parameters
if EEEM^v is infeasible
    $UB = \text{EEEM}^v$
end if
else
    $LB = \text{EEEM}^v$
end else
Step 4: Termination
if $UB - LB < \epsilon$
    return $\text{EEEM}^* = LB$
end if
else
$v = v + 1$
$\text{EEEM}^v = (UB + LB)/2$
Go to Step 2
end else

Appendix D

Reduction Method Alternatives

1) Furthest Distance to Closest Point (FDCP): FDCP starts with an arbitrary scenario (e.g., scenario 1). Then, for each scenario outside the reduced scenario set, it calculates the minimum of the distances (using $l = 2$ norm) between the scenario and scenarios in the reduced scenario set. Next, it adds the scenario with the highest minimum distance to the reduced scenario set. The method repeats the scenario adding steps until the reduced scenario set, Ω', reaches a predetermined size (K).

2) Furthest Distance to Centroid (FDC): FDC starts with an arbitrary scenario (e.g., scenario 1). Then, for each scenario outside the reduced scenario set, it calculates the distance (using $l = 2$ norm) between the scenario and the centroid of the reduced scenario set. The centroid can be an imaginary scenario. Next, it adds the scenario with the highest distance to the reduced scenario set. The method repeats the scenario adding steps iteratively until the reduced scenario set, Ω', reaches a predetermined size (K).

Algorithm 3 Furthest Distance to Closest Point

$\Omega^{'} = \{1\}$
while $|\Omega^{'}| < K$
  $d_{max} = -\infty$
  $\omega_{max} = NULL$
  for $\forall \omega \in \Omega \setminus \Omega^{'}$
    $d_{min} = \infty$
    for $\forall \omega^{'} \in \Omega^{'}$
      if $||\omega, \omega^{'}||_2 < d_{min}$
        $d_{min} = ||\omega, \omega^{'}||_2$
      end if
    end for
    if $d_{min} > d_{max}$
      $d_{max} = d_{min}$
      $\omega_{max} = \omega$
  end for
  Update $\Omega^{'} = \Omega^{'} \bigcup \omega_{max}$
end while

Algorithm 4 Furthest Distance to Centroid

$\Omega^{'} = \{1\}$
while $|\Omega^{'}| < K$
  $d_{max} = -\infty$
  $\omega_{max} = NULL$
  Generate the centroid $\hat{\omega}$ such that $t^{\hat{\omega}} = \{t \in \mathbb{R}^{|I|} | t_i = \frac{\sum_{\omega \in \Omega^{'}} t_i^{\omega}}{|\Omega^{'}|}\}$
  for $\forall \omega \in \Omega \setminus \Omega^{'}$
    if $||\omega, \hat{\omega}||_2 > d_{max}$
      $d_{max} = ||\omega, \hat{\omega}||_2$
      $\omega_{max} = \omega$
  end for
  Update $\Omega^{'} = \Omega^{'} \bigcup \omega_{max}$
end while

3) Minimum Total Coverage Distance Model (CDM): Coverage distance of scenario $\omega \in \Omega$ is the minimum of the distances between scenario ω and all the scenarios in the reduced set. The MIP model below forms a reduced scenario set with a predetermined size (K) that minimizes the total coverage distance across all scenarios. For distance calculations, $l = 2$ norm is used. At an optimal solution, $u_\omega = 1$ if ω is in the reduced set, and $v_{\omega,\omega'} = 1$ if $\omega^{'}$ is the closest point to ω in the reduced set.

$$
\begin{aligned}
\min \quad & \sum_{\omega\in\Omega}\sum_{\omega'\in\Omega} ||\omega,\omega^{'}||_2 v_{\omega,\omega'} & \\
\text{s.t.} \quad & \sum_{\omega\in\Omega} u_\omega = K & \\
& \sum_{\omega'\in\Omega} v_{\omega,\omega'} = 1 & \forall\omega\in\Omega \\
& v_{\omega,\omega'} \leq \; u_{\omega'} & \forall\omega,\omega^{'}\in\Omega \\
& u_\omega \in \{0,1\} & \forall\omega\in\Omega \\
& v_{\omega,\omega'} \in \{0,1\} & \forall\omega,\omega^{'}\in\Omega
\end{aligned}
$$

4) Maximum Total Diversity Model (TDM): Diversity distance of scenario $\omega \in \Omega'$ is the minimum of the distances between scenario ω and $\omega^{'}$, $\forall\omega^{'} \in \Omega^{'}\setminus\{\omega\}$. Letting δ_ω denote the diversity distance of scenario ω, the MIP model below forms a reduced scenario set, $\Omega^{'}$, with a predetermined size (K), that maximizes the total diversity distance across all scenarios in this reduced scenario set. For distance calculations, $l = 2$ norm is used.

$$
\begin{aligned}
\min \quad & \sum_{\omega \in \Omega} \delta_\omega \\
\text{s.t.} \quad & \sum_{\omega \in \Omega} u_\omega = K \\
& \delta_\omega \leq \ ||\omega, \omega^{'}||_2 u_\omega + M(1 - u_{\omega^{'}}) && \forall \omega, \omega^{'} \in \Omega, \omega^{'} \neq \omega \\
& \delta_\omega \geq 0 && \forall \omega \in \Omega \\
& u_\omega \in \{0, 1\} && \forall \omega \in \Omega
\end{aligned}
$$

5) MIP formulation to minimize Wasserstein distance with l norm: The MIP model below forms a reduced scenario set, $\Omega^{'}$, with a predetermined size (K) that minimizes the Wasserstein distance between Ω and $\Omega^{'}$. $l = 1$ or $l = 2$ norms are used in the objective function. Wasserstein distance is the distance between two probability distributions. In the model, $\pi_{\omega\omega^{'}}$ represents the fraction of the probability of scenario ω transferred to scenario $\omega^{'}$.

$$
\begin{aligned}
\min \quad & \sum_{\omega \in \Omega} \sum_{\omega^{'} \in \Omega} ||\omega, \omega^{'}||_l \pi_{\omega,\omega^{'}} \\
\text{s.t.} \quad & \sum_{\omega \in \Omega} u_\omega = K \\
& \sum_{\omega^{'} \in \Omega} \pi_{\omega,\omega^{'}} = 1 && \forall \omega \in \Omega \\
& \sum_{\omega^{'} \in \Omega} \pi_{\omega,\omega^{'}} = \frac{|\Omega|}{K} u_{\omega^{'}} && \forall \omega \in \Omega \\
& u_\omega \in \{0, 1\} && \forall \omega \in \Omega \\
& \pi_{\omega,\omega^{'}} \geq 0 && \forall \omega, \omega^{'} \in \Omega
\end{aligned}
$$

6) Local Search to minimize Wasserstein distance with l norm: The method starts with an arbitrary reduced scenario set, $\Omega^{'}$ with a predetermined size (K). In our implementation, we choose the first K scenarios. Then, it checks for each pair $\{\omega', \omega\} : \omega' \in \Omega', \omega \in \Omega \setminus \Omega'$ whether the Wasserstein distance between Ω and $(\Omega' \setminus \{\omega'\}) \cup \{\omega\}$ is less than the Wasserstein distance between Ω and Ω'. Then, it chooses the best pair and updates the reduced scenario set by switching scenarios ω' and ω. The method repeats these steps and stops after checking all such pairs.

Algorithm 5 Local Search to minimize Wasserstein Distance

$\Omega' = \{1, ..., K\}$
d_{best} = Wasserstein distance between Ω and Ω'
$d_{change} = d_{best}$
while $d_{change} \leq d_{best}$
  $d_{change} = \min_{\omega' \in \Omega', \omega \in \Omega \setminus \Omega'}$ Wasserstein distance between Ω and $(\Omega' \setminus \{\omega'\}) \cup \{\omega\}$
  $out, in = \arg\min_{\omega' \in \Omega', \omega \in \Omega \setminus \Omega'}$ Wasserstein distance between Ω and $(\Omega' \setminus \{\omega'\}) \cup \{\omega\}$
  if $d_{change} < d_{best}$
    $d_{best} = d_{change}$
    $\Omega' = (\Omega' \setminus \{out\}) \cup \{in\}$
  else
    STOP

Appendix E

Conditional Selection Method

Algorithm 6 Conditional Selection Method at iteration v

if EEEM^v is infeasible
  UB =EEEM^v
  if $|\text{EEEM}^v - \text{EEEM}^{v-1}| > \epsilon$
    EEEM^{v+1} $= (UB + LB)/2$
  end if
end if
else
  if $|\text{EEEM}^v - \text{EEEM}^{v-1}| > \epsilon$
    LB_1 =EEEM^v
    EEEM^{v+1} =EEEM^v $+ \epsilon$
  end if
  else
    LB_2 =EEEM^v
    EEEM^{v+1} $= (LB_1 + UB)/2$
  end else
end else
$LB = max(LB_1, LB_2)$

Appendix F

Extensive Formulation

$$
\begin{aligned}
\max \quad & \text{EEEM} && && \text{(F.1)} \\
\text{s.t.} \quad & b_{ij}{+}b_{ji}{=}1 && \forall i,j{\in}I, j>i && \text{(F.2)} \\
& a_j{\geq}a_i{-}M\big(1{-}b_{ij}\big) && \forall i,j{\in}I, j{\neq}i && \text{(F.3)} \\
& a_i{\leq}H && \forall i && \text{(F.4)} \\
& \sum_{n\in N} x^{\omega}_{in}{=}1 && \forall i{\in}I, \forall\omega{\in}\Omega && \text{(F.5)} \\
& \sum_{c\in C} y^{\omega}_{ic}{=}1 && \forall i{\in}I, \forall\omega{\in}\Omega && \text{(F.6)} \\
& a_j{+}w^{\omega}_j{\geq}a_i{+}w^{\omega}_i{-}M\big(1{-}b_{ij}\big) && \forall i,j{\in}I, j{\neq}i, \forall\omega{\in}\Omega && \text{(F.7)} \\
& a_j{+}w^{\omega}_j{\geq}a_i{+}w^{\omega}_i{+}s{-}M\big(3\text{-}b_{ij}\text{-}x^{\omega}_{in}\text{-}x^{\omega}_{jn}\big) && \forall i,j{\in}I, j{\neq}i, \forall n{\in}N, \forall\omega{\in}\Omega && \text{(F.8)} \\
& a_j{+}w^{\omega}_j{\geq}a_i{+}w^{\omega}_i{+}s{+}t^{\omega}_i\text{-}M\big(3\text{-}b_{ij}\text{-}y^{\omega}_{ic}\text{-}y^{\omega}_{jc}\big) && \forall i,j{\in}I, j{\neq}i, \forall c{\in}C, \forall\omega{\in}\Omega && \text{(F.9)} \\
& O^{\omega}_n{\leq}L && \forall n{\in}N, \forall\omega{\in}\Omega && \text{(F.10)} \\
& O^{\omega}_n{\geq}a_i{+}w^{\omega}_i{+}s{+}t^{\omega}_i{-}H{-}M\big(1{-}x^{\omega}_{in}\big) && \forall i{\in}I, n{\in}N, \forall\omega{\in}\Omega && \text{(F.11)} \\
& q^{\omega}_i{\geq}w^{\omega}_i{-}\eta^{\omega} && \forall i{\in}I, \forall\omega{\in}\Omega && \text{(F.12)} \\
& \eta^{\omega}{+}\frac{1}{|I|(1{-}\alpha^{\omega})}\sum_{i\in I} q^{\omega}_i{\leq}\tau && \forall\omega{\in}\Omega && \text{(F.13)} \\
& \text{EEEM}{\leq}\alpha^{\omega} && \forall\omega{\in}\Omega && \text{(F.14)}
\end{aligned}
$$

$$y_{11}^{\omega}=1 \qquad \forall \omega \in \Omega \tag{F.15}$$

$$y_{ic}^{\omega}=0 \qquad \forall i \in I, \forall c \in C, 1<i<c, \forall \omega \in \Omega \tag{F.16}$$

$$a_i \leq M\theta_{1i} \qquad \forall i \in I \tag{F.17}$$

$$\sum_{j,j\neq i} b_{ij}-(|I|-|N|-1)\leq M(1-\theta_{1i}) \qquad \forall i \in I \tag{F.18}$$

$$a_i-(m-1)s \leq M\theta_{mi} \qquad \forall i \in I \tag{F.19}$$

$$(m-1)s-a_i \leq M\theta_{mi} \qquad \forall i \in I \tag{F.20}$$

$$\sum_{j,j\neq i} b_{ij}-(|I|-(m-1)|N|-1)\leq M\theta_{mi} \qquad \forall i \in I, m=2,...,k+1 \tag{F.21}$$

$$(|I|-(m)|N|)-\sum_{j,j\neq i} b_{ij} \leq M\theta_{mi} \qquad \forall i \in I, m=2,...,k \tag{F.22}$$

$$(|I|-|C|)-\sum_{j,j\neq i} b_{ij} \leq M\theta_{k+1,i} \qquad \forall i \in I \tag{F.23}$$

$$s+t_{min}-a_i \leq M\theta_{k+2,i} \qquad \forall i \in I \tag{F.24}$$

$$(|I|-|C|)-\sum_{j,j\neq i} b_{ij} \leq M(1-\theta_{k+2,i}) \qquad \forall i \in I \tag{F.25}$$

$$\sum_{m=1,...,k+2} \theta_{mi}=k+1 \qquad \forall i \in I \tag{F.26}$$

$$\sum_{i\in I} \theta_{mi}=|I|-|N| \qquad m=1,...,k \tag{F.27}$$

$$\sum_{i\in I} \theta_{k+1,i}=|I|-l \tag{F.28}$$

$$\sum_{i\in I} \theta_{k+2,i}=|C| \tag{F.29}$$

$$O_n^{\omega}-O_{n+1}^{\omega}\geq 0 \qquad \forall n \in N, n\neq |N|, \forall \omega \in \Omega \tag{F.30}$$

$$O_1^{\omega}+H\geq\frac{\sum_i t_i^{\omega}+(|I|+\sum_{m=1}^{k}(m\text{-}1)|N|+m(|C|\text{-}k|N|))s}{|C|} \qquad \forall\omega\in\Omega \quad \text{(F.31)}$$

$$\text{EEEM}\in[0,1) \quad \text{(F.32)}$$

$$\alpha^{\omega}\in[0,1) \qquad \forall\omega\in\Omega \quad \text{(F.33)}$$

$$b_{ij}\in\{0,1\} \qquad \forall i,j\in I, j\neq i \quad \text{(F.34)}$$

$$a_i\in\mathbb{Z}^+ \qquad \forall i \quad \text{(F.35)}$$

$$x_{in}^{\omega}\in\{0,1\} \qquad \forall i\in I, n\in N, \forall\omega\in\Omega \quad \text{(F.36)}$$

$$y_{ic}^{\omega}\in\{0,1\} \qquad \forall i\in I, c\in C, \forall\omega\in\Omega \quad \text{(F.37)}$$

$$w_i^{\omega}, q_i^{\omega}\geq 0 \qquad \forall i\in I, \forall\omega\in\Omega \quad \text{(F.38)}$$

$$\eta^{\omega}\geq 0 \qquad \forall\omega\in\Omega \quad \text{(F.39)}$$

$$O_n^{\omega}\geq 0 \qquad \forall n\in N, \forall\omega\in\Omega \quad \text{(F.40)}$$

$$\theta_{mi}\in\{0,1\} \qquad m=1,...,k+2, \forall i\in I \quad \text{(F.41)}$$

Appendix G

Dissimilarity with EEEM Schedules

Table G.1: $l = 1$ Distance of schedules to EEEM schedule (based on appointment times)

	Instance										
Schedule	1	2	3	4	5	6	7	8	9	10	Average
SPT	134	55	52	185	180	185	164	150	159	104	136.80
LPT	231	496	255	198	157	143	167	230	159	289	232.50
Var	141	120	26	127	141	118	176	150	174	39	121.20
CoV	231	496	255	174	154	143	145	226	140	276	224
MAD	141	65	22	185	141	118	176	150	186	39	122.30
Baseline	524	292	331	546	600	636	643	535	640	464	521.10
Minmax	8	5	4	24	49	0	27	6	0	92	21.50

Table G.2: $l = 1$ Distance of schedules to EEEM schedule (based on patient positions)

	Instance										
Schedule	1	2	3	4	5	6	7	8	9	10	Average
SPT	22	18	26	20	20	22	22	22	20	16	20.80
LPT	32	22	28	24	26	26	24	22	26	24	25.40
Var	20	10	20	28	22	24	34	12	16	10	19.60
CoV	34	30	26	26	24	24	20	22	20	26	25.20
MAD	18	8	22	30	20	22	34	16	14	10	19.40
Baseline	22	18	26	20	20	22	22	22	20	16	20.80
Minmax	0	0	0	2	12	0	0	14	0	12	4.00