Personalized Data-Driven Learning & Optimization: Theory and Applications to Healthcare

by

Esmaeil Keyvanshokooh

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Industrial and Operations Engineering)
in The University of Michigan
2021

Doctoral Committee:

Professor Mark P. Van Oyen, Co-Chair
Associate Professor Cong Shi, Co-Chair
Professor Wallace J. Hopp
Assistant Professor Yuekai Sun

Esmaeil Keyvanshokooh
keyvan@umich.edu
ORCID iD: 0000-0001-9634-3806

DEDICATION

To my parents and my spouse,

for their unconditional love and continuous support.

ACKNOWLEDGEMENTS

I would like to sincerely thank my advisor, Professor Mark P. Van Oyen for his incredible guidance, innumerous support and encouragement, and for everything he has taught me throughout the tenure of my Ph.D. study, without which this dissertation would not have been completed. He has also been more than a thesis advisor for me, providing me with advice on a broader level. I have learned so much from him through the years: he has taught me how to learn, how to teach, how to be a good colleague and collaborator. I appreciate the freedom I was given to explore my interests while enjoying tremendous intellectual support.

I have also been greatly privileged to work closely under the supervision of Professor Cong Shi. I owe a debt of gratitude to him. His guidance and support have been unparalleled: he has been always a source of inspiration for me, generous in his time, and truly supportive, especially when needed most. He has contributed enormously in shaping and advancing my thinking and research direction. By being inspirational, he has always lifted my motivations when I have encountered barriers.

I would also like to thank Professor Wallace Hopp for his encouragement and support. I have been flattered to have the opportunity of learning interesting healthcare topics from him. I am very much thankful to Professor Yuekai Sun for his comments and support for my Ph.D. dissertation. My special gratitude goes to Professor Brian Denton for letting me teach at IOE and always being supportive to me. I appreciate the supports from Seth Bonder Foundation and Rackham Pre-Doctoral Fellowship, which have been critical in successfully finishing this dissertation. I have also been greatly benefitted to work closely with my colleagues, Pooyan Kazemian and Mohammad Zhalechian, on different research topics.

I am especially grateful to my parents, Tooba and Ebrahim, who have raised me to be the person I am today. I am forever thankful to you for your unwavering support throughout the years, and instilling in me a passion for learning and pursuing my goals. I would also like to thank my wonderful in-laws, Zahra and Ahmad, for their kindness and continual support.

A very special thank you to my spouse, Elnaz, for her love, unconditional support and all her positive thoughts. Without your love and support, I would have never come this far. Thank you for being there for me every step of the way and giving me the endurance and strength to arrive at this milestone. Words cannot express my gratitude for having you.

I owe gratitude to my wonderful and inspiring siblings for always being there for me since day one and despite the distance. Thanks for helping me learn from you, and being my best friends. Thank you for loving me at my worst, just as much as my best.

Thank you to the many people in Ann Arbor who have made this place home. I have made some of the best friends I could ever ask for - Mohammad Ali, Delaram, Mohsen, Sajedeh, Mehrdad, Niloofar, Amin, Neda, Mahnaz, and Iman. Thank you for creating many fond memories that I will treasure. Without you, I would not have been able to enjoy my life here in Ann Arbor throughout these years. Friday night gatherings have been a highlight of my weeks.

TABLE OF CONTENTS

LIST OF FIGURES

Figure

LIST OF TABLES

Table

ABSTRACT

This dissertation is broadly about developing new personalized data-driven learning and optimization methods with theoretical performance guarantees for three important applications in healthcare operations management and medical decision-making. In these research problems, we are dealing with longitudinal settings, where the decision-maker needs to make multi-stage personalized decisions while collecting data in-between stages. In each stage, the decision-maker incorporates the newly observed data in order to update his current sys-tem's model or belief, thereby making better decisions next. This new class of data-driven learning and optimization methods indeed learns from data over time so as to make efficient and effective decisions for each individual in real-time under dynamic, uncertain environ-ments. The theoretical contributions lie in the design and analysis of these new predictive and prescriptive learning and optimization methods and proving theoretical performance guarantees for them. The practical contributions are to apply these methods to resolve un-met real-world needs in healthcare operations management and medical decision-making so as to yield managerial and practical insights and new functionality.

In Chapter II, we focus on chronic diseases that are the leading cause of mortality and disability worldwide, requiring the surveillance and monitoring of each patient to assess disease progression and determine if an appropriate intervention for that individual is warranted. In many cases, it is a challenge to determine the most effective treatment. Even when a suitable treatment is identified, dosing it correctly remains a significant challenge because the proper dosage depends on the individual. This involves adaptively learning a personal-ized disease progression control model conditional on patient-specific contextual information. We formulate this as a new contextual multi-armed bandit under a two-dimensional control with a nested structure, which sequentially selects the best treatment and corresponding dosage based on contextual information. With the goal of minimizing disease progression risk, we develop a joint contextual learning and optimization algorithm that integrates the strength of contextual bandit with online convex optimization techniques. We prove that this algorithm admits a sub-linear regret, which is tight up to a logarithmic factor. We also derive some general technical results that are of independent interest. The effectiveness of our methodology is illustrated using case data on patients with type 2 diabetes.

In Chapter III, we studied a fundamental class of online resource allocation problems, where a heterogeneous stream of patient referrals arrives one at a time with a declared urgency-based reward for receiving service from one of the heterogeneous providers. Upon arrival of a referral, the system chooses both a provider and a time for service over a multi-day horizon subject to provider capacity. For this problem, we developed a new class of online optimization algorithms based on a primal-dual approach, and called it the Heterogeneous Online Optimization Procedure with Budgeted Overtime (HOOP-BOT). The online policies derived by HOOP-BOT offer urgency-sensitive appointment visits for patients while achieving high utilization. They are robust to future information, easy-to-implement, and efficient to compute, allow for heterogeneity in both urgency-based rewards and service times, and admit a theoretical competitive ratio, guaranteeing the worst-case performance. Using data from a partner hospital, our online policies greatly outperform benchmark policies.

In Chapter IV, we study a coordinated clinic and surgery appointment scheduling problem in a surgical suite. Our aim is to provide timely access to care by coordinating clinic and surgery visits to ensure that patients can see a surgeon in the clinic and schedule their surgery within a maximum wait time target. There are different types of uncertainty including the number of appointment visits, whether a patient requires surgery, and surgery durations. We develop an Integrated Multi-stage Stochastic and Distributionally Robust Optimization (IMSDRO) methodology to determine the optimal clinic and surgery dates such that the access constraints are satisfied, and the overtimes are minimized. This approach integrates a multi-stage stochastic program with a distributionally robust optimization approach to simultaneously incorporate multiple types of uncertainties by including stochastic scenarios for appointment request arrivals and ambiguity sets for surgery durations. Several new transformations are introduced to turn the nonlinear program to a tractable one, and a constraint generation algorithm is developed to solve it. We propose a data-driven rolling horizon procedure to facilitate implementation. Using case data, we show that our approach significantly improves access delay times compared with the current practice.

CHAPTER I

Introduction

The rapid growth of information and accessibility to big data provide an unprecedented opportunity to shift toward personalized data-driven decision-making, thereby tailoring decisions to individuals. In today's era of personalization, every decision-maker needs to deploy personalized data or contextual information to optimize the decisions and achieve better outcome. For example, healthcare delivery presents many decision support opportunities for personalized or precision treatment choices based on patient bio-markers and clinical history. In marketing, these same methods can increase the click-through rate through ads and promotions tailored to the user's demographics and interests. In these scenarios, it is often the case that patients or customers will arrive sequentially; therefore, the decision-maker requires to make a sequence of personalized decisions in real-time or online fashion and improve these decisions with incremental information. These online personalized decision-making paradigms (i) adaptively learn a statistical model that predicts a user-specific outcome for each available decision as a function of the user's observed contextual information, and (ii) harness this model to make optimize personalized decisions for subsequent users.

This dissertation studies a fundamentally new way of thinking about online data-driven decision-making, and presents significant advances to both theoretical and practical aspects of this paradigm in the context of healthcare delivery systems and chronic disease treatment management. We develop a set of new data-driven decision-making approaches, building on statistical machine learning, online convex optimization, multi-armed bandit, stochastic optimization, and distributionally robust optimization theories. They offer substantial opportunities to help and protect our communities from wide range of chronic diseases and healthcare delivery systems. This dissertation is presented in a multiple manuscript format as independent academic papers. We provide three distinct frameworks to facilitate the presentation of our online data-driven learning and optimization approaches. In the following, we summarize each chapter and its contributions.

Chapter II - Contextual Learning with Online Convex Optimization: Theory and Applications to Chronic Diseases: Modeling disease progression based on real-world data is a challenge due to patient-level heterogeneity, uncertain nature of the disease, multiple potential interventions, irregularity and noise of tests. One must choose from multiple treatments available to control the disease progression. For each treatment, the dosage selection is unique to that selected treatment, which makes the optimization of dosing more complex. We call this the nested decision-making. For this setting, we develop a personalized medicine approach using a contextual multi-armed bandit to sequentially tailor the treatment selection to each patient based on the bandit feedback of prior medical history and biomarkers (i.e., contextual information). This requires a judicious trade-off between selecting the hitherto best treatment (exploitation) and choosing an alternative treatment to collect more information (exploration).

There is, however, a novel contribution to the literature in part because unlike most bandit settings, we have a two-dimensional control with a nested structure, having dosage choice nested under the treatment selection. Doctors choose a treatment for a patient and decide on which dosage of the chosen treatment to prescribe. Inspired by this setting, we introduced a new contextual multi-armed bandit (MAB) with a two-dimensional nested control. This model can be seen as an extended MAB, where the decision-maker must opt for which arm to pull (treatment) and how far to pull the selected arm (dosage). The difficulty lies in the fact that the reward function is not jointly convex in these two decisions. To resolve this issue, we proposed a joint contextual learning and optimization algorithm, which deploys a posterior sampling approach to find the best treatment for an individual while performing stochastic sub-gradient descent optimization procedure (S-SGD) to obtain the best corresponding dosage. This integration of the contextual MAB with online convex optimization necessitates the development of new high-probability concentration bounds. Indeed, we proved a performance guarantee, a regret, for this algorithm. Deriving this regret involves bounding: (i) estimation loss for estimating unknown parameters, (ii) contextual bandit loss for learning the optimal treatment, and (iii) S-SGD sub-optimality loss for learning the optimal dosage of each treatment. Adding these losses does not yield the regret; instead, we merged them through a new bridging technique argument.

For societal impact and insight, we implemented our theory using data on high blood-pressure (BP) patients. Doctors lack strong evidence on which medication and dosage to prescribe for these patients to lower BP as little as possible while achieving a safe target. Notably, our method outperforms the current practice to find the optimal regimen. Our theory does allow for many other applications with two-dimensional decision-making arising in operations management (e.g., joint assortment selection and pricing).

Chapter III - Online Advance Scheduling with Overtime: A Primal-Dual Approach: This chapter was motivated by a dramatic expansion of new outpatient space being built for a partner hospital over the next couple of years. Accordingly, we addressed the situations, where historical distributions for demand are either not available (e.g., new service systems) or face unpredictable demand change. This needs building robustness and online (real-time) allocation capability into online appointment platforms.

In this chapter, we study a fundamental online resource allocation problem in service operations in which a heterogeneous stream of arrivals that varies in service times and rewards makes service requests from a finite number of servers or providers. This is an online adversarial setting in which nothing more is known about the arrival process of customers (i.e., no knowledge of the arrival process). Each server has a finite regular capacity but can be expanded at the expense of overtime cost. Upon arrival of each customer, the system chooses both a server and a time for service over a scheduling horizon subject to capacity constraints. The system seeks easy-to-implement online policies that admit a competitive ratio (CR), guaranteeing the worst-case relative performance for these policies.

On the theoretical side, we propose online algorithms with theoretical CRs for the problem described above. On the practical side, we investigate the real-world applicability of our methods and models on appointment-scheduling data from a partner hospital. We develop new online primal-dual approaches for making not only a server-date allocation decision for each arriving customer, but also an overtime decision for each server on each day within a horizon. We also derive a competitive analysis to prove a theoretical performance guarantee. Our online policies are (i) robust to future information, (ii) easy-to implement and extremely efficient to compute, and (iii) admitting a theoretical CR. Comparing our online policy with the optimal offline policy, we obtain a CR that guarantees the worst-case performance of our online policy. We evaluate the performance of our online algorithms by using real appointment scheduling data from a partner hospital. Our empirical results show that the proposed online policies perform much better than their theoretical CR, and also outperform the pervasive first-come-first-served and nested threshold policies by a large margin.

Chapter IV - Coordinated and Priority-based Surgical Care: An Integrated Distributionally Robust Stochastic Optimization Approach: This chapter is motivated by a collaboration with Mayo Clinic that seek for achieving timely access to surgery. Their limited surgical capacity along with inherent uncertainty in both arrival processes and surgery times lead to barriers to both timely access to care and efficient resource utilization. To tackle this issue, we introduced the idea of care coordination across different stages of patient care to ensure proper follow-up treatments and prevent health complications.

In this chapter, we developed a new Integrated Multi-stage Stochastic Distributionally

Robust Optimization (IMSDRO) methodology to achieve this care coordination. While guaranteeing priority-based clinical and surgical access delay targets, this approach offers (i) the optimal clinic date, and (ii) the optimal surgery date (the need for surgery is realized in the clinic visit). The objective is to minimize the average of clinical and surgical overtimes. To make the IMSDRO approach implementable in practice, we developed a data-driven rolling horizon procedure. This allows practitioners to make optimal use of data that is revealed as time progresses, and adjust their decisions on a rolling basis to use the realization of uncertainty in arrivals, surgical need, and surgery duration. Our methodology is not limited to a particular setting and can be applied to other service operations industries where access to the service matters.

Using case data, Our coordinated stochastic-robust policy improves the surgical access times by about 160%, on average, compared to the current policy used by our partner hospital. Unlike the current policy that operates based on a first-come first-serve idea, intuitively our policy often defers the surgery of low-priority patients in order to preserve the near future capacity to serve high-priority patients that may arrive later. This approach helps meet the desired service level with minimum overtime.

Chapter V - Conclusions and Future Research: The works presented in Chapters II-IV make contributions into three important parts of personalized data-driven learning and optimization with applications in management of chronic disease progression, online appointment scheduling platforms, and healthcare delivery. In Chapter V, we summarize some of our most important contributions. We also highlight areas of future research that could expand on this work.

CHAPTER II

Contextual Learning with Online Convex Optimization: Theory and Applications to Chronic Diseases [1]

2.1 Introduction

Chronic diseases are reversible or irreversible conditions that are at risk of progression throughout a long period of time (often years or decades). They place tremendous burdens on patients, their carers, and the healthcare delivery system. Common chronic diseases include diabetes, cancer, heart, and kidney diseases. They are the leading causes of mortality and the largest sources of cost to the U.S. healthcare system, accounting for more than 80% of the nation's healthcare costs [57]. A better understanding of *disease progression* is essential in early diagnosis and long-term treatment. However, modeling chronic disease progression based on real-world evidence is a challenging task due to patient-level heterogeneity, uncertain nature of chronic diseases, multiple potential interventions, and tests that may be noisy and irregular in timing.

There are usually *multiple treatments* available to control the risk of chronic disease progression. Even when a suitable treatment is identified, *dosing* it optimally remains a significant challenge, because the correct dosage may be as important as the correct treatment ([105] and [21]). For example, for patients with warfarin, incorrect dosage may lead to adverse outcomes, such as stroke (if the dose is too low) and internal bleeding (if the dose is too high) ([35]). Effective dosing of a selected treatment should take into account *both* over-dosing and under-dosing outcomes. Under-dosing results in poor disease control (e.g., uncontrolled progression and symptoms), while over-dosing incurs excessive side-effects (e.g.,

[1] Under Revision at Management Science as Keyvanshokooh, E., Zhalechian, M., Shi, C., Van Oyen, M. P., Kazemian, P. (2020), Contextual Learning with Online Convex Optimization: Theory and Applications to Chronic Diseases.

kidney complications, anemia risk, bloating, toxicity, and diarrhea).

In spite of all the above-mentioned difficulties in modeling chronic disease progression, the growing availability of patient-specific data from electronic medical records provides medical professionals with easy access to more patient biomarkers and clinical history. This supports a personalized treatment choice with a corresponding dosage that is optimized over those patients who have been prescribed that treatment in the past. Unlike one-size-fits-all medicine, personalized medicine tailors such decisions to individual patients based on their demographic, medical history, clinical tests, and biomarkers (i.e., contextual information). To develop a personalized approach in real-time, one can (i) *adaptively* learn a model that predicts a patient-specific outcome for each available decision as a function of the patient's observed contextual information, and (ii) deploy such a predictive model for subsequent new patients to optimally adjust treatment decision in a personalized way and corresponding dosage decision that is optimized over the prior history of patients seen. We call this approach the "*joint contextual learning and optimization*" approach.

In this setting, a sequential decision-making process with *bandit* feedback, often modeled as a multi-armed bandit (MAB), is necessitated. This implies that clinicians only acquire feedback from their selected treatment and dosage, and do not observe counterfactual from other possible decisions that they could have made. This hurdle inspires a judicious trade-off between *exploration* and *exploitation*. Doctors may select the hitherto best treatment based on prior experience administrating that treatment to similar patients or extrapolation from similar treatments (i.e., exploitation). Yet, such a decision could be sub-optimal since it is made based on the predictions derived from a limited number of observed samples. Alternatively, they may decide to collect more information on an alternative treatment that is not presently seen as the best (i.e., exploration). Because this alternative may eventually prove to be the best one, seeking this new information prevents doctors from locking into a misperception caused by a lack of data. There is, however, a salient challenge in handling the exploration-exploitation dilemma for disease management. Unlike most bandit settings, our problem has a *two-dimensional control* with a *nested structure*, which includes (i) treatment and (ii) dosage that is unique to the chosen treatment. Modeling this nested decision feature spurs a new technical development in the MAB theory.

To date, a considerable number of studies have been done on predicting chronic disease progression risk and identifying the optimal treatment regimen by developing *offline statistical learning* algorithms (see e.g., [30] and [164]). Both offline and online statistical learning algorithms rely on historical data to obtain estimations; however, *online statistical learning* algorithms leverage the advantage of *adaptive learning*. Balancing the exploration-exploitation trade-off, they collect new information adaptively to learn as quickly as possible,

rather than just waiting to use a large historical data set. Online statistical learning algorithms take advantage of the current beliefs to make treatment regimen decisions (exploitation) while learning more about poorly estimated treatment regimens (exploration). This ensures that the treatment regimen is offered based on the particular needs of the individual, and reduces the risk of offering a poorly understood treatment regimen to a large patient population.

In light of all the above discussions, we will address the following two issues in this paper: (i) how one can provide a personalized joint contextual learning and optimization approach for chronic diseases, which adaptively learns a predictive model that is able to select both treatment and dosage with the aim of reducing the disease progression risk, and (ii) how to design this approach to provide a good performance guarantee. These questions motivate us to integrate online learning and online convex optimization methods.

2.1.1 Main Results and Contributions

Our main contribution is the introduction of a new contextual multi-armed bandit model with a two-dimensional nested control, and the development and analysis of a new joint contexutal learning and optimization algorithm for it that we call the *contextual bandit with stochastic sub-gradient descent* (S-SGD Bandit) algorithm. This algorithm makes a *two-dimensional nested decision* (i.e., the best treatment and corresponding dosage) for each individual given the observed contextual information. The objective is to reduce the disease progression risk, which is a common way to quantify a chronic disease. Below, we shall summarize our main results and contributions.

(1) Contextual multi-armed bandit with online convex optimization. The salient feature of our method, in contrast to prior studies, is the integration of the *contextual multi-armed bandit* and *online convex optimization*, which we shall elaborate below.

In our setting, based on the patient's contextual information, the doctor needs to jointly decide the treatment choice and its corresponding dosage. Inspired by this setting, we introduce a new contextual MAB with a two-dimensional nested control (see §2.2). This can be viewed as an extended MAB, where the decision-maker must select *which arm to pull* (treatment) as well as *how far to pull* the selected arm (dosage). The difficulty lies in the fact that the reward function is *not jointly convex* in these two decision variables. On a high level, the proposed algorithm resolves this difficulty by deploying a posterior sampling approach to find the best treatment for each individual while performing online stochastic sub-gradient descent (S-SGD) for each selected treatment to obtain the best corresponding dosage.

Most online learning algorithms in the literature do not capture the control with two

levels of decision making; neither do they address nested decisions (e.g., [21], [22], [48], [67], and [5]). Our integration of contextual MAB with online convex optimization necessitates the development of new high-probability concentration bounds. First, we have two bandit feedback models: a disease progression risk and a treatment-outcome metric. We leverage properties of the reward function to unify our high-probability bounds for estimating the unknown parameters of these two models, which yields an *estimation loss* (Proposition 1). Second, we develop a high-probability bound on a *contextual bandit loss* incurred due to learning the optimal treatment (Proposition 2). Third, to integrate the S-SGD optimization procedure with contextual bandit, we derive a high-probability bound for S-SGD instead of common almost-sure convergence results (Proposition 3). This bound for the sub-Gaussian S-SGD is based on a martingale difference sequence argument due to dependence among dosage decisions over time, and it is of independent interest beyond this paper. Lastly, unlike most bandit settings, in practice patient feedback *cannot* be revealed immediately after a treatment regimen is prescribed. For instance, the delay in observing patient feedback ranges from two weeks to one month in our case study. We introduce an *on the fly* strategy for both contextual bandit and S-SGD to deal with the *delayed bandit feedback* under *stochastic delay*. This strategy updates the system based only on the realized information.

We derive a performance guarantee for the S-SGD Bandit algorithm using the notion of *Bayesian regret*, which is cumulative expected loss of our online policy compared to the clairvoyant optimal policy obtained under full information. Bounding the Bayesian regret requires bounding (i) *estimation loss* incurred due to the online estimation of unknown parameters (Proposition 1), (ii) *contextual bandit loss* for learning the optimal treatment (Proposition 2), and (iii) *S-SGD sub-optimality loss* for learning the optimal dosage of each treatment (Proposition 3). It is important to note that simply adding these three losses does not obtain the final regret. Instead, we merge these three loss bounds via a new *bridging technique* to yield a Bayesian regret $\widetilde{\mathcal{O}}\big((d+K)\sqrt{T}(1+\sqrt{N_{\max}})\big)$, where d is the context dimension, K is the number of arms, and $N_{\max}$ is the maximum number of unrealized feedbacks over T periods (Theorem II.1 and Corollary II.2). This regret is provably optimal up to a logarithmic factor (see discussions in §2.4.2). In §2.4.2, we also discuss how our regret bound improves the existing theoretical result on the both contextual and Lipschitz bandits.

We extend this result on the online convex optimization case to the one that does not necessarily have access to a gradient oracle at any point in the dosage decision space, which is called bandit convex optimization (see §2.4.7). Thus, we also develop a high-probability loss bound for Bandit SGD (B-SGD) (Proposition 4), and obtain a regret of $\widetilde{\mathcal{O}}\big(T^{3/4}\big)$ for another algorithm that we call the *B-SGD Bandit algorithm*, in which B-SGD is used instead

of S-SGD (Proposition 5). In general, we observe that in terms of final regret, the contextual bandit and estimation losses outweigh the S-SGD sub-optimality loss for the S-SGD Bandit algorithm, whereas the B-SGD sub-optimality loss outweighs the contextual bandit and estimation losses for the B-SGD Bandit algorithm.

(2) Application to chronic disease progression management for blood pressure control. In collaboration with medical researchers, we assess our model/algorithm using a recent ACCORD clinical trial data on high Blood Pressure (BP) patients with type 2 diabetes at high risk for cardiovascular disease in §2.5. Several medications should be introduced in multiple stages, referred to as first-, second-, and third-line treatments. The recommended course for the first two medications is fairly well established by clinical guidelines [14]. However, the clinical guidelines *lack* the third-line medication and its dosage. Our goal is to learn the optimal third-line treatment regimen to minimize the risk of having high systolic BP. We demonstrate that our methodology outperforms the existing clinical practices and benchmark policies to find the optimal treatment regimen. We also obtain four important clinical implications in §2.5.4.

(3) General contextual learning and optimization framework for personalized decision-making. Although our joint contextual learning and optimization framework is motivated by a fundamental medical decision-making problem, it delivers a general cutting-edge method to many other applications with *two-dimensional nested* decision-making arising in operations management. They include joint pricing and assortment selection, joint inventory control and pricing, and joint inventory control and vehicle routing, among others.

2.1.2 Literature Review

This work is relevant to two research domains and streams of literature discussed below.

Chronic disease progression control. Chronic diseases are becoming the most pressing health issue worldwide, constituting a considerable portion of yearly deaths [122]. Seminal works in disease progression modeling were published in the early 1990s (e.g., [77] and [87]), but the field has flourished recently. To control disease progression, most models are derived from average responses to treatment in patient populations. [164] presented a fused-group Lasso model to predict the disease progression and select biomarkers predictive of the Alzheimer's progression. [156] proposed a probabilistic disease progression model using unsupervised learning techniques to predict a continuous-time disease progression trajectory from a set of patients' records. [30] built offline statistical models (regularized linear regression, random forests and support vector machines) based on historical data for advanced gastric cancer to design new combination chemotherapy regimens. [155] introduced a recurrent neural networks model to predict the Alzheimer's progression for a next visit. [138] developed a

deep learning model for chronic disease progression, intervention recommendation and future risk prediction. [31] presented an optimization-based machine learning method to identify new targets of existing treatments to increase the treatable cancer patients.

Another trend is toward adaptive clinical trial design ([51] and [26]), which uses data accumulated during a trial to decide whether the trial should be modified or stopped based on conclusive evidence on the effectiveness of a treatment. For example, [49] and [50] studied Bayesian adaptive trial designs under delayed outcome observations and multiple correlated treatments, respectively. [103] optimized decisions about opening new test sites, setting the patient recruitment rate, and stopping decisions at interim stages of an adaptive trial design. [12] developed a Bayesian adaptive trial design to incorporate data from multiple outcomes. Unlike our paper, this research stream has the goal of establishing the safety and effectiveness that is expected to apply to the vast majority within a sub-population, satisfying specific inclusion and exclusion criteria [99].

Several approaches have been proposed to identify personalized treatment regimens. They include sequential multiple assignment randomized trials ([130] and [11]), stochastic control theory ([84] and [97]), doubly robust estimation ([131] and [36]), Markov decision processes ([9], [146], and [16]) and reinforcement learning ([139], [140], and [104]) to identify the optimal personalized treatment regimen from historical data. While these methods rely on offline observational data and cannot guide the data collection procedure, our work differs from these studies, in that it has adaptive data collection with each arriving patient to adaptively modify aspects of the treatment regimen as soon as possible.

Multi-armed bandit. MAB is an online learning framework for making sequential decisions over time when the effect of each action on the outcome is uncertain. The agent selects an action at each step with the aim of maximizing the expected cumulative rewards of the selected actions. The Upper Confidence Bound (UCB) and Thompson Sampling (TS) are the two common algorithms for solving MAB. UCB is based on the principle of optimism in the face of uncertainty and selects the action with the highest optimistic (upper bound) estimate of the expected reward. TS is a randomized Bayesian algorithm, which assumes that there is a prior distribution on the unknown parameter of the reward distribution. The idea is to randomly sample from the posterior distribution of the reward of each action, and then select the action with the highest sampled reward. Contextual MAB is an extension of MAB in which the reward of each action depends on the context.

There has been a growing interest in the development of personalized medicine methods using MAB. [8] formulated a MAB model to study how changing treatment decisions during the course of a trial can achieve better health outcomes. [132] developed a continuous-time MAB model to construct personalized treatments for chronic diseases by continuously

monitoring patients' states and infrequent health events, such as disease relapses and flare-ups. [127] presented a non-stationary MAB model, which optimizes the selection of messages that should be sent to each individual so as to improve physical activity and adherence. None of these works consider contextual patient information for making decisions.

Turning to the theoretical literature on the contextual MAB, [15] introduced the first algorithm for the linear contextual MAB. [53], [142], [52], and [1] improved this algorithm by presenting other UCB-based algorithms. [68] designed a UCB-based algorithm for the generalized linear contextual MAB, which was improved by [108]. [6] and [144] proposed TS-based algorithms for the linear contextual MAB.

[76] presented the OLS bandit algorithm for a two-armed bandit with i.i.d. context vectors. [21] presented the LASSO bandit for high-dimensional contexts. [22] developed an exploration-free greedy algorithm for contextual MAB. [20] studied a contextual MAB with cross-learning, where the learner also learns the reward that would have been achieved by choosing the same action under different contexts. [48] proposed a tune sliding window-UCB algorithm for the non-stationary MAB. [82] studied a general approach to analyze stochastic linear MABs. Note that none of these studies have treated contextual MAB under a two-dimensional control.

2.1.3 Paper Organization and General Notation

The remainder of this paper is organized as follows. We formulate the problem in §2.2 and introduce the proposed algorithm in §2.3. We carry out a non-asymptotic regret analysis in §2.4. In §2.5, we provide a case study using type 2 diabetes mellitus data. Finally, we conclude our paper in §2.6.

All vectors are column vectors. For any vector $x \in \mathbb{R}^n$, x^T denotes its transpose, and $[x]_k$ presents its k^{th} element. The Euclidean norm and weighted norm of the vector x are $\|x\| = \sqrt{x^T x}$ and $\|x\|_M = \sqrt{x^T M x}$, respectively. The inner product of two vectors is defined as $\langle x, y \rangle = \sum_{i=1}^{n} x_i y_i$. The determinant of a matrix M is denoted by $\det(M)$, and I denotes the identity matrix. For a symmetric positive definite matrix V, we define λ_{min} as the smallest eigenvalues of V. For two symmetric matrices V and M, $V \succeq M$ means that $V - M$ is positive semidefinite. We use $\mathbb{1}(\cdot)$ as the indicator function. The projection operator for projection of a point $x \in \mathbb{R}^n$ onto a convex set $\mathcal{C}$ is defined by $\mathbf{Proj}_{\mathcal{C}}(x) \triangleq \arg\min_{y \in \mathcal{C}} \|x - y\|$.

2.2 Contextual Multi-Armed Bandit with Two-Dimensional Control

We formally define the proposed contextual multi-armed bandit with a two-dimensional control in the context of a personalized disease progression control system model in this section. Consider a finite time horizon of length T. We define $\mathcal{T} = \{1, \ldots, T\}$ as the set of time periods.

Two-dimensional control and context. Unlike most bandit settings, we have a *two-dimensional* control with a nested structure, which includes (i) treatment and (ii) dosage. The doctors should choose a treatment from a set of possible treatments for a patient, and also decide a corresponding dosage of the selected treatment. This can be viewed as an extended multi-armed bandit problem where the decision maker must choose *which arm* (treatment) to pull as well as *how far to pull* the selected arm (i.e., the dosage can be thought of as how far to pull the selected arm). Indeed, each arm has a control (dosage) that affects the arm's performance.

There are two bandit feedbacks (main outcome and sub-outcome). In particular, in the context of chronic diseases, we need to minimize the *disease progression risk* as the main outcome, through optimizing both treatment and its corresponding dosage for an individual. The disease progression risk often depends on a treatment and dosage effectiveness measure that we shall call the *treatment-dosage sub-outcome* metric. For an example for this setting, refer to the case study in §2.5.

We denote $\mathcal{K} = \{1, \ldots, K\}$ to be the set of actions, and each action/treatment $k \in \mathcal{K}$ corresponds to a K-dimensional *action vector* $\phi_k^{\mathcal{A}}(t)$, where the subscript k in $\phi_k^{\mathcal{A}}(t)$ corresponds to the selected treatment (e.g., consider $\phi_k^{\mathcal{A}}(t) = (\mathbb{1}_{k=A}, \mathbb{1}_{k=B}, \mathbb{1}_{k=C})^T$, where A, B, and C are three different treatments, and only one of them is selected). Also, each action $k \in \mathcal{K}$ corresponds to a K-dimensional *dosage vector* $y_k(t)$, where k^{th} element belongs to $[\Delta_k^{LB}, \Delta_k^{UB}]$ and all other elements are zero.

At each time period t, we observe the contextual information associated with a patient. Each patient t has a two-part context vector $(\phi^{\mathcal{X}}(t), \psi^{\mathcal{X}}(t))$, where $\phi^{\mathcal{X}}(t) \in \mathbb{R}^{d_1}$ and $\psi^{\mathcal{X}}(t) \in \mathbb{R}^{d_2}$, which is picked by an oblivious adversary and does not necessarily come from any fixed distribution. The difference between these two parts of the context is that $\phi^{\mathcal{X}}(t)$ directly affects the disease progression risk, while $\psi^{\mathcal{X}}(t)$ indirectly affects the disease progression risk through its effect on the treatment-outcome metric. For instance, for patients with chronic kidney disease, a part of the context (e.g., cholesterol and triglycerides) affects the disease progression risk (main outcome) through its effect on blood pressure (sub-outcome), and another part of the context (e.g., potassium and serum creatinine) affects the disease

progression risk directly. We define $\Phi_k(t) := (\phi^{\mathcal{X}}(t)^T, \phi_k^{\mathcal{A}}(t)^T)^T$ as the *feature vector*, which concatenates the context vector $\phi^{\mathcal{X}}(t)$ and the action vector $\phi_k^{\mathcal{A}}(t)$. We define $\Psi_k(t) := (\psi^{\mathcal{X}}(t)^T, \phi_k^{\mathcal{A}}(t)^T)^T$ as another *feature vector*, which concatenates the context vector $\psi^{\mathcal{X}}(t)$ and the action vector $\phi_k^{\mathcal{A}}(t)$ (see Figure 2.1). For ease of notation, we denote $\Upsilon_k(t) = (\Psi_k(t)^T, y_k(t)^T)^T$ as the augmented feature vector, which includes the dosage vector as well.

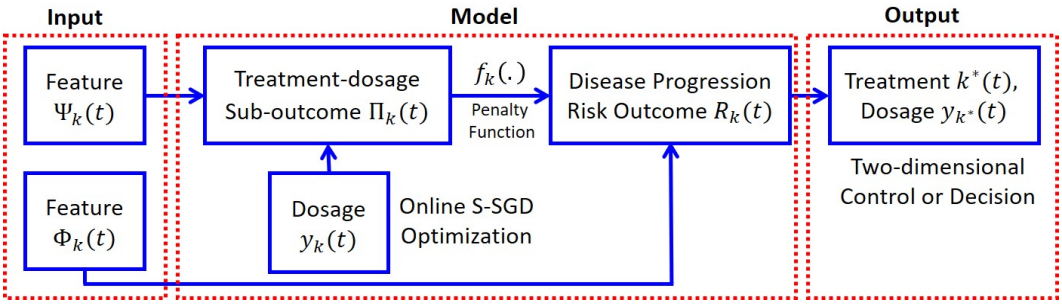


Figure 2.1: The illustration of the personalized disease progression control system modeled by a contextual multi-armed bandit with a two-dimensional control for making treatment and corresponding dosage decisions.

Patient reward (disease progression risk). Choosing each action k yields an uncertain binary reward $R_k(t)$ (main outcome), where $R_k(t) = +1$ corresponds to success (i.e., the disease is *not-progressed*) and $R_k(t) = -1$ corresponds to failure (i.e., the disease is *progressed*). If we choose action $k \in \mathcal{K}$ at time t, it yields an uncertain binary reward with the following expected value:

$$\mathbb{E}\big[R_k(t)\big] = \sigma\Big(\Phi_k^T(t) \cdot \theta - f_k\big(\Psi_k(t), y_k(t);\ \pi\big)\Big), \tag{2.1}$$

where $\sigma(\cdot) : \mathbb{R} \to \mathbb{R}_+$ is a logistic function, $\theta \in \mathbb{R}^{d_1+K}$ is the unknown (true) parameter corresponding to the feature vector $\Phi_k(t)$, and f_k is the dosage penalty function with the unknown (true) parameter $\pi \in \mathbb{R}^{d_2+2K}$ defined in (2.2). The error $\xi_k(t) = R_k(t) - \sigma\big(\Phi_k^T(t) \cdot \theta - f_k(\Psi_k(t), y_k(t); \pi)\big)$ is a 1-sub-Gaussian variable (see Definition 1) conditional on all the previous realized feature vectors, dosages, and rewards. The set $\mathcal{H}_t$ is the history of all the observed information available at the beginning of time period t. We assume that there are deterministic sets for $\Phi_k(t)$ and $\Psi_k(t)$. Without loss of generality, we also assume that $\|\Phi_k(t)\| \leq 1$, $\|\Psi_k(t)\| \leq 1$, $\|\theta\| \leq c_\theta$ and $\|\pi\| \leq c_\pi$.

Definition 1. *A real-valued random variable ξ is λ-sub-Gaussian if* $\mathbb{E}[e^{t\xi}] \leq e^{\lambda^2 t^2/2}, \ \forall t \in \mathbb{R}$.

This definition implies that $\mathbb{E}[\xi] = 0$ and $\mathrm{Var}[\xi] \leq \lambda^2$. Many distributions are sub-Gaussian, including any bounded and centered distribution, and Gaussian distribution.

Dosage penalty function. When the decision-maker decides on a dosage of a selected treatment k for a patient, there is often a trade-off between under-dosing outcome (e.g., poor disease control) and over-dosing outcome (e.g., side effects of medication). We model this trade-off by a dosage penalty function $f_k(.) : \mathbb{R}^{d_2+2K} \to \mathbb{R}$ for each treatment $k \in \mathcal{K}$ as follows:

$$f_k\big(\Psi_k(t), y_k(t);\ \pi\big) = \alpha_k \underbrace{\Big(\Psi_k^T(t)\cdot\omega + y_k^T(t)\cdot\tau - q\Big)^+}_{\text{over-dosing}} + \beta_k \underbrace{\Big(q - \Psi_k^T(t)\cdot\omega - y_k^T(t)\cdot\tau\Big)^+}_{\text{under-dosing}}, \tag{2.2}$$

where $\pi = (\omega^T, \tau^T)^T \in \mathbb{R}^{d_2+2K}$ is the unknown (true) vector parameter corresponding to the feature vector $\Psi_k(t)$ and the dosage vector $y_k(t)$, respectively. Also, $\alpha_k, \beta_k \in \mathbb{R}_+$ are the known penalty rates for over-dosing and under-dosing, respectively, and $q \in \mathbb{R}_+$ denotes the known threshold between over-dosing and under-dosing regions. This is the target for the treatment-dosage sub-outcome that clinicians want to control for. For example, doctors usually target a systolic BP of 120 mmHg under the intensive high BP therapy for patients with type 2 diabetes (see §2.5).

Note that the dosage penalty function $f_k(.)$ evaluates how bad over-dosing or under-dosing is for each patient. The stochasticity of this function comes from the uncertainty in the feature information and an unknown parameter π. Although we know the structure of the dosage penalty function, there is an unknown parameter π, which hinders us from minimizing this function to find the optimal dosage. This issue is addressed in §2.4.5.

Treatment-dosage sub-outcome. If we choose a treatment k with corresponding dosage $y_k(t)$ for patient t with feature information $\Psi_k(t)$, it yields an uncertain *treatment-dosage sub-outcome* $\Pi_k(t)$ with the following expected value:

$$\mathbb{E}\big[\Pi_k(t)\big] = \Psi_k^T(t)\cdot\omega + y_k^T(t)\cdot\tau, \tag{2.3}$$

where $\pi = (\omega^T, \tau^T)^T \in \mathbb{R}^{d_2+2K}$ is the unknown (true) vector parameter corresponding to the feature vector $\Psi_k(t)$ and the dosage vector $y_k(t)$, respectively. The error $\zeta_k(t) = \Pi_k(t) - \Psi_k^T(t)\cdot\omega - y_k^T(t)\cdot\tau$ is a λ-sub-Gaussian variable (see Definition 1) conditional on the previous realized feature vectors, dosages, and treatment-dosage sub-outcomes.

Note that this metric can be measured at each patient visit in many practical settings. For instance, for patients with type 2 diabetes (see §2.5), systolic blood pressure, which is easily measured, can be presumed as a treatment-dosage sub-outcome, because maintaining a normal systolic blood pressure is of paramount important for reducing the risk of cardiovascular events and mortality.

Furthermore, it should be pointed out that unlike most classical contextual MABs, we incorporate *delayed feedback* for the treatment-dosage sub-outcome and reward. For example, in our case study in §2.5, true patient feedback is often realized within two to four weeks on average after prescribing a treatment with a specified dosage to a patient. We assume that $\{D(t)\}_{t=1}^{T}$ are i.i.d. non-negative random variables with mean μ_D that satisfy the following regularity condition:

$$\mathbb{P}\big(D(t) - \mu_D \geq m\big) \leq \exp\Big(-\frac{m^{p+1}}{2\sigma_D^2}\Big), \tag{2.4}$$

for some $p \geq 0$ and $\sigma_D > 0$. This assumption includes the most common delay patterns in practice (e.g., $D(t)$ is an exponential delay when $p = 0$ and $D(t)$ is a sub-Gaussian delay when $p = 1$).

The goal of our contextual bandit with two-dimensional control model is to maximize the cumulative expected rewards (or the probability of preventing a disease progression) over a finite horizon of length T under delayed feedback while learning the patient-specific rewards.

2.3 S-SGD Bandit Algorithm

We present the proposed joint learning and optimization algorithm for the above-described model. This algorithm synergizes both the contextual bandit and online convex optimization techniques and provides a personalized disease progression control system. We describe the high-level intuition of the proposed algorithm in §2.3.1. In §2.3.2, we provide the detailed steps.

2.3.1 Main Ideas of the S-SGD Bandit Algorithm

There are often several treatment options with different dosages for controlling the patient's disease progression. However, there is uncertainty about the impact of each treatment and its corresponding dosage on controlling disease progression. In the proposed algorithm, patients can be characterized by a unique set of characteristics. The best treatment is then selected based on these characteristics, and the corresponding dosage is optimized over the prior history of patients seen.

We do not know the true distribution of patient rewards; hence, we need to learn the distribution of unknown parameters θ and π while making treatment and dosage decisions adaptively. We put some Gaussian prior distributions for these parameters based on some prior beliefs. The initial results from a pilot study can be utilized to construct such reasonable prior beliefs. In the next step, the algorithm provides an estimate for the expected reward

(i.e., probability of avoiding disease progression) after observing the patient's contextual information. This estimate depends on these two unknown parameters, which need to be learned based on the available information. The exploration-exploitation trade-off should be addressed properly. For example, according to the parameter estimates in the early stage, we may incorrectly conclude that one treatment with a corresponding dosage are not an appropriate option for a patient with a certain medical history, and subsequently may not be able to identify this incorrect belief without choosing a different treatment and corresponding dosage for a very similar patient. Inspired by the idea of *posterior sampling*, the algorithm assumes posterior distributions over the unknown parameters θ and π, and then randomly samples from these distributions. The intuition behind this sampling is to balance the exploration-exploitation trade-off. If the algorithm only uses the mean of the posterior distribution in each time period as an estimate for the unknown parameters, it exploits the current belief about the unknown parameter, and so there is no possibility for exploring the alternative choices. However, there is no guarantee that the estimated value is optimal. Thus, we add an *exploration phase* by taking a random sample from the posterior distributions. We have an *exploitation phase* as well in which we choose the policy that provides the maximum expected reward.

The algorithm leverages the new revealed feedbacks (i.e., whether the disease is progressed as well as the treatment-dosage sub-outcome) to update the posterior distribution of the unknown parameters and also the dosage of each treatment. In practice, unlike most bandit settings, the patient feedback cannot be revealed immediately after a treatment with a corresponding dosage are prescribed. For instance, in our case study in §2.5, the patient feedback is realized from two to four weeks on average. The algorithm uses an *on the fly* strategy to deal with the *delayed feedback* with *stochastic* delay. It employs the information of the patient to whom a treatment with a specified dosage is already prescribed only if the patient's feedback is revealed up to the current time.

For dosage selection, the algorithm starts with a safe dosage from a range of possible dosages for each treatment option. It then deploys an *online stochastic sub-gradient descent* optimization procedure in which the dosage of each treatment is updated by leveraging only the revealed patients' feedbacks. This optimization procedure involves taking the (noisy) derivative of the dosage penalty function for all patients with newly realized feedbacks. For each such patient, this is obtained by checking whether the patient has an under-dosing or over-dosing for the prescribed treatment. It should be noted that the treatment selection is personalized, but there is a universal best dosage for each treatment that is learned over time. Finally, by exploiting the new revealed feedbacks at each time period, the algorithm updates the current belief about the posterior distributions of the unknown vector parameters θ and

π. This updating procedure is performed using an online Bayesian regression based on a Laplace approximation (see Appendix F for details).

2.3.2 Description of TS-based S-SGD Bandit Algorithm

Initialization. Initialize a safe, arbitrary dosage vector $y_k(1)$ for each treatment $k \in \mathcal{K}$, where its k^{th} element belongs to $[\Delta_k^{LB}, \Delta_k^{UB}]$ and all other elements are zero. Initialize the step size η_k.

Parameters. Let m_ℓ^1 and $(q_\ell^1)^{-1}$ be the mean and variance of the Gaussian prior distribution for the ℓ-th element of θ vector. Let u^1 and $(P^1)^{-1}$ be the mean and covariance matrix of the Gaussian prior distribution for π vector. All these parameters can be initialized based on some prior beliefs.

Main Loop. We proceed in time periods $\mathcal{T} = \{1, \ldots, T\}$, which correspond to epochs of patient arrivals. There are six main steps in each time period $t \in \mathcal{T}$, described below.

Step 1 (Context Information). Observe the context information $(\phi^{\mathcal{X}}(t), \psi^{\mathcal{X}}(t))$ of patient t.

Step 2 (Sampling). In this step, random samples are drawn from the posterior distributions of parameters. These samples will be used in estimating the unknown parameters in the next step.

(2a) Draw a sample $[\tilde{\theta}(t)]_\ell$ from the posterior normal distribution $[\theta(t)]_\ell \sim \mathcal{N}(m_\ell^t, (q_\ell^t)^{-1})$ for each corresponding element $\ell \in \{1, \ldots, d_1 + K\}$ of the feature vector.

(2b) Draw a sample $\tilde{\pi}(t)$ from the posterior normal distribution $\pi(t) \sim \mathcal{N}(u^t, (P^t)^{-1})$ for the corresponding feature and dosage vectors.

Step 3 (Policy Optimization and Implementation). Having the samples $\tilde{\theta}(t)$, and $\tilde{\pi}(t) = (\tilde{\omega}(t), \tilde{\tau}(t))$, choose the treatment $k(t)$ with the corresponding dosage $y_k(t)$ for patient t, where

$$k(t) = \arg\max_{k \in \mathcal{K}} \left\{ \sigma\Big(\Phi_k^T(t) \cdot \tilde{\theta}(t) - f_k\big(\Psi_k(t), y_k(t); \tilde{\pi}(t)\big) \Big) \right\}.$$

Step 4 (Feedback Observation). For each treatment $k \in \mathcal{K}$, obtain $\mathcal{S}_k(t)$ as the set of time-stamps with *new realized* feedbacks (i.e., rewards and treatment-dosage sub-outcomes) at time period t, which is calculated by $\mathcal{S}_k(t) = \mathcal{M}_k(t+1) - \mathcal{M}_k(t)$, where the set $\mathcal{M}_k(t)$ contains the time-stamps with realized feedbacks by the end of time period $t-1$ corresponding to treatment k.

Step 5 (Online Stochastic Sub-Gradient Descent). Update and calculate the following:

(5a) Calculate $\widetilde{\nabla} f_k(s) = [\tilde{\tau}(t)]_k \Big(\alpha_k \mathbb{1}\{\Pi_k(s) > q\} - \beta_k \mathbb{1}\{\Pi_k(s) < q\}\Big)$ for each realized treatment-dosage sub-outcome $\Pi_k(s)$ with time-stamp $s \in \mathcal{S}_k(t)$ at time period t for each $k \in \mathcal{K}$.

(5b) Obtain the next period's dosage for each treatment $k \in \mathcal{K}$ by

$$[y_k(t+1)]_k = \mathbf{Proj}_{[\Delta_k^{LB}, \Delta_k^{UB}]}\Big([y_k(t)]_k - \eta_k \sum_{s \in \mathcal{S}_k(t)} \widetilde{\nabla} f_k(s)\Big).$$

Step 6 (Belief Updating). We leverage the patients' bandit feedback (i.e., whether the disease is progressed, and the treatment-dosage sub-outcome) whose time-stamp is in $\mathcal{S}_k(t)$ for each treatment k to update the posterior distributions of θ and π vector parameters.

(6a) Solve the following optimization problem,

$$\hat{\rho} \triangleq \arg\max_{\rho} \ \frac{1}{2} \sum_{\ell=1}^{d_1+K} q_\ell^t \, ([\rho]_\ell - m_\ell^t)^2 + \sum_{k \in \mathcal{K}} \sum_{s \in \mathcal{S}_k(t)} \log\left(1 + e^{-R_k(s)\,(\rho^T \Phi_k(s) - f_k(\Upsilon_k(s); \tilde{\pi}(t)))}\right).$$

(6b) Update the mean and variance of the posterior distribution for θ and π, respectively,

$$m^{t+1} = \hat{\rho},\ q_\ell^{t+1} = q_\ell^t + \frac{e^{-(\hat{\rho}^T \Phi_k(t) - f_k(\Upsilon_k(t); \tilde{\pi}(t)))}}{(1 + e^{-(\hat{\rho}^T \Phi_k(t) - f_k(\Upsilon_k(t); \tilde{\pi}(t)))})^2} \Big([\Phi_k(t)]_\ell\Big)^2, \forall \ell \in \{1, \ldots, d_1 + K\}$$

$$P^{t+1} = P^t + \Psi_k(t) \cdot \Psi_k^T(t),\ e^{t+1} = e^t + \sum_{k \in \mathcal{K}} \sum_{s \in \mathcal{S}_k(t)} \Psi_k(s) \cdot \Pi_k(s),\ u^{t+1} = (P^{t+1})^{-1} e^{t+1}.$$

It is worth noting that the UCB-based S-SGD Bandit variant (Algorithm 2) of the above proposed algorithm is presented in Appendix D.

2.4 Theoretical Performance Analysis and Discussions

We derive a non-asymptotic (i.e., finite-time) performance guarantee for the S-SGD Bandit algorithm by proving a regret bound. We start by defining the performance measure in §2.4.1. We then state the main theoretical result, and position it in the related literature in §2.4.2. In §2.4.3, we provide a roadmap for our regret analysis, which is derived in §2.4.4, §2.4.5, and §2.4.6. In §2.4.7, we extend our result to the bandit convex optimization and present the regret of the B-SGD Bandit algorithm.

2.4.1 Performance Measure

We evaluate the performance of the S-SGD Bandit algorithm in terms of the expected regret (see Definition 2), where the expectation is taken over the prior distribution of reward function. This is also called the *Bayesian regret* since it represents the *Bayes risk* ([142] and [144]). The Bayesian regret has two main advantages: (i) it allows for any arbitrary prior distribution for the unknown parameters, and (ii) it bridges between the TS and UCB methods, which provides the opportunity to leverage some appealing theoretical properties of UCB methods in deriving the Bayesian regret. Our benchmark is the clairvoyant optimal policy, which knows the true parameters (i.e., it knows the expected reward and treatment-outcome metric) and chooses the optimal treatment given each individual and the optimal dosage for each prescribed treatment.

Definition 2 (**Bayesian Regret**). *Given the unknown vector parameters θ and π, the T-time period regret of our learning and optimization algorithm is defined by*

$$\textbf{Regret}(T,\, \theta,\, \pi) = \mathbb{E}\big[OPT - ALG|\, \theta,\, \pi\big],$$

where OPT and ALG are the total rewards of the clairvoyant optimal policy and our online policy, respectively, over T time periods. The conditional expectation is taken over the random realization of the rewards given θ and π, and the random samples drawn from the posterior distributions. Bayesian regret over the T time periods is then defined by

$$\textbf{BayesianRegret}(T) = \mathbb{E}\big[\textbf{\textit{Regret}}(T,\, \theta,\, \pi)\big],$$

where the expectation is taken over the prior distributions of θ and π.

2.4.2 Main Theoretical Results

Below, we first state our main theoretical results and position them in the related literature.

Theorem II.1 (**Bayesian Regret of the TS-based S-SGD Bandit Algorithm**). *The Bayesian regret of the TS-based S-SGD Bandit algorithm over finite time horizon T is as follows:*

(a) under the immediate feedback (no delay) setting,

$$\mathcal{O}\left(\sqrt{T}\left((d_1+K)\log\left(T/(d_1+K)\right)+(d_2+K)\log\left(T/(d_2+K)\right)\right)\right),$$

(b) under the delayed feedback with stochastic delay setting,

$$\mathcal{O}\Big(\sqrt{T}\,(d_1+K)\Big(\log\big(T/(d_1+K)\big)+\sqrt{N_{\max}\log(N_{\max}/(d_1+K))}\Big)+ \\ \sqrt{T}\,(d_2+K)\Big(\log\big(T/(d_2+K)\big)+\sqrt{N_{\max}\log(N_{\max}/(d_2+K))}\Big)\Big),$$

where K is the number of actions (treatments), d_1 and d_2 are the dimensions of context vectors. Also, $N_{\max}$ is the maximum number of unrealized feedbacks by the time period T, that is upper bounded by $2\mu_D+\tilde{\sigma}\Big(\sqrt{2\log T}+\sqrt{2\log\big(1/\delta\big)+c'(\tilde{\sigma}\sqrt{2\log T}+1)}\Big)+c$, with probability $1-\delta$, where $c=2\,\tilde{\sigma}^2\log\,(2\sigma_D^2+1)+1$, $c'=2\log(2\sigma_D^2+1)$, and $\tilde{\sigma}=\sigma_D\sqrt{p+2}$.

Corollary II.2 (**Regret of the UCB-based S-SGD Bandit Algorithm**). *The UCB-based S-SGD Bandit algorithm has the regret of the same order as the Bayesian regret (Theorem II.1).*

Discussions of the Main Results. It is worthwhile positioning our theoretical results (for the contextual multi-armed bandit with a two-dimensional decision space) in the related literature.

First, we shall relate to the literature on linear *contextual bandits* with d-dimensional context vectors (but with only one-dimensional decision space). [53] and [1] present UCB-based policies with $\widetilde{\mathcal{O}}(d\sqrt{T})$ worst-case regret bound, which is minimax optimal up to logarithmic factors, as proved by [53] for infinite number of arms. [52] achieve a lower bound $\Omega(\sqrt{dT})$ for finite number of arms. [144] derive a $\widetilde{\mathcal{O}}(d\sqrt{T})$ Bayesian regret bound for TS-based policies, which matches the regret bound of [53] for UCB-based policies. [6] and [2] obtain a $\widetilde{\mathcal{O}}(d^{3/2}\sqrt{T})$ frequentist regret bound for a variant of TS-based policies, which is far from the optimal rate by a factor of $\sqrt{d}$. [68] develop a $\mathcal{O}(d\log^{3/2}(T)\sqrt{T})$ regret bound for generalized linear contextual bandits. We cannot replicate the algorithms and regret analyses of any of these studies for our problem, which spurs new methodological innovations. This is because we have (i) a two-dimensional decision space, and (ii) two bandit feedbacks. The highest order term of our regret does not depend on delays and is tight (optimal) up to a logarithmic factor compared to the lower bound of [53]. It is also within a factor $\sqrt{d}$ of the lower bound of [52]. The feedback delay impacts the second highest order term of the regret by a factor of $\sqrt{N_{\max}\log(N_{\max})}$, where $N_{\max}$ is upper bounded by a logarithmic quantity in T.

Second, we shall also relate our paper to the literature on *Lipschitz bandit* with two-dimensional decision space. Note that this line of literature does not consider contextual information (while our setting does). Ignoring the contextual information, our problem can be viewed as a two-dimensional Lipschitz bandit problem where one dimension is treatment, and the other one is the corresponding dosage. For this two-dimensional Lipschitz bandit

setting, [101] and [38] present online policies with the optimal regret of $\mathcal{O}\big(T^{3/4}\big)$. However, in our problem the treatment desicisons are discrete, and given a treatment, the reward function is convex in the dosage level, and hence has a richer structure than the Lipschitz bandit. The balk of our analysis is to utilize this richer structure to lower the regret from $\mathcal{O}\big(T^{3/4}\big)$ to a regret of order $\mathcal{O}\big(\sqrt{T}\log T\big)$. To achieve this regret improvement, we obtain high-probability bounds for estimation, contextual bandit, and S-SGD losses, and then merge them through a new *bridging technique*.

We also note that our theoretical result applies to any general convex and sub-differentiable dosage penalty function (e.g., news-vendor-like dosage function considered in our model) that its gradient can be observed in each iteration, and our techniques can be used in other settings.

2.4.3 Roadmap for Proving the Main Theoretical Result

We provide a road-map for proving the main theoretical result (Theorem II.1). We first introduce the following notations for the expected reward function of any treatment k at time period t under (i) estimated parameter $(\hat{\theta}(t), \hat{\pi}(t))$ and a sub-optimal dosage $y_k(t)$, (ii) true parameter (θ, π) and a sub-optimal dosage $y_k(t)$, and (iii) true parameter (θ, π) and the optimal dosage y_k^*, respectively:

$$\begin{aligned}
\hat{\mathbf{V}}_k(t) &:= \sigma\Big(\Phi_k^T(t)\cdot\hat{\theta}(t) - f_k\big(\Psi_k(t), y_k(t);\ \hat{\pi}(t)\big)\Big),\\
\mathbf{V}_k(t) &:= \sigma\Big(\Phi_k^T(t)\cdot\theta - f_k\big(\Psi_k(t), y_k(t);\ \pi\big)\Big),\\
\mathbf{V}_k^*(t) &:= \sigma\Big(\Phi_k^T(t)\cdot\theta - f_k\big(\Psi_k(t), y_k^*;\ \pi\big)\Big).
\end{aligned}$$

The Bayesian regret can be then derived by the following *bridging technique* (see Figure 2.2):

$$\begin{aligned}
\textbf{BayesianRegret}(T) = \mathbb{E}\Big[\sum_{t=1}^{T}\Big(\mathbf{V}_{k^*}^*(t) - \mathbf{V}_k(t)\Big)\Big] &= \mathbb{E}\Big[\sum_{t=1}^{T}\Big(\mathbf{V}_{k^*}^*(t) - \mathbf{V}_k^*(t)\Big)\\
&+ \sum_{t=1}^{T}\Big(\mathbf{V}_k^*(t) - \hat{\mathbf{V}}_k(t)\Big) + \sum_{t=1}^{T}\Big(\hat{\mathbf{V}}_k(t) - \mathbf{V}_k(t)\Big)\Big].
\end{aligned}$$

In the above decomposition, the first term reflects the loss due to finding the optimal treatment (termed *contextual bandit loss*), the second term reflect the loss due to finding the optimal dosage of a treatment (termed *S-SGD sub-optimality loss*), and the last one corresponds to the online estimation of unknown parameters (termed *estimation loss*).

Figure 2.2 outlines the main steps of the regret analysis. Proposition 1 derives a high-

probability confidence bound on the online estimation of the expected reward for each selected treatment k at each time period t, which results in bounding the estimation loss $\mathbb{E}\big[\sum_{t=1}^{T}\big(\hat{\mathbf{V}}_k(t)-\mathbf{V}_k(t)\big)\big]$. Proposition 2 develops a high-probability confidence bound on the contextual bandit loss $\mathbb{E}\big[\sum_{t=1}^{T}\big(\mathbf{V}^*_{k^*}(t)-\mathbf{V}_k(t)\big)\big]$ incurred due to learning the optimal treatment decision k^*. Lastly, a high-probability confidence bound on the S-SGD sub-optimality loss $\mathbb{E}\big[\sum_{t=1}^{T}\big(\hat{\mathbf{V}}_k(t)-\mathbf{V}_k(t)\big)\big]$ incurred due to finding the optimal dosage decision for any treatment option k is obtained by using Propositions 3 and 1.

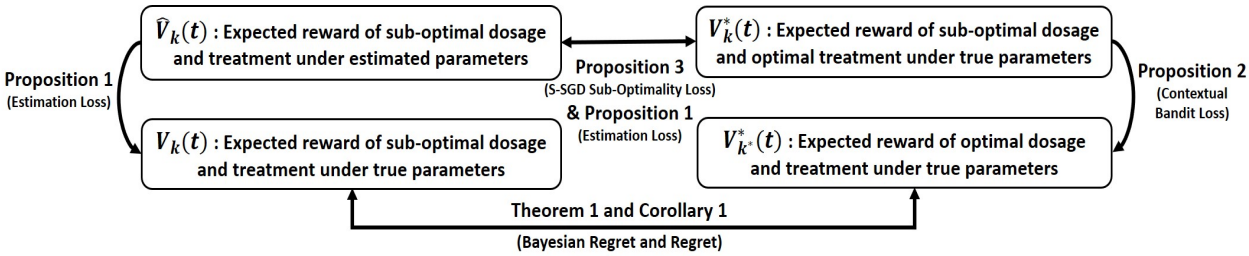


Figure 2.2: The outline for deriving the regret of the proposed joint contextual learning and optimization algorithms (S-SGD and B-SGD Bandit algorithms).

2.4.4 Theoretical Results on Estimation and Contextual Bandit Losses

In this section, we first establish a high-probability bound on the estimation loss between true and estimated expected reward or $|\mathbf{V}_k(t)-\hat{\mathbf{V}}_k(t)|$ of each selected treatment k with any dosage in Proposition 1. Using this result, Proposition 2 bounds the contextual bandit loss.

Proposition 1 (**Estimation Loss for Expected Reward of each Treatment**). For any t and $\delta > 0$, the following loss bound on the difference between the true and estimated expected rewards under each selected treatment k holds with probability at least $1-2\delta$:

$$\Big|\sigma\Big(\Phi_k^T(t)\cdot\theta - f_k\big(\Psi_k(t),y_k(t);\pi\big)\Big)-\sigma\Big(\Phi_k^T(t)\cdot\hat{\theta}(t) - f_k\big(\Psi_k(t),y_k(t);\hat{\pi}(t)\big)\Big)\Big| \tag{2.5}$$
$$\leq \frac{1}{2c_\sigma}\left\|\Phi_k(t)\right\|_{V_t^{-1}}\left(\sqrt{2\log\left(\frac{\det(V_t)^{1/2}\det(\gamma I)^{-1/2}}{\delta}\right)}+\sqrt{2(d_1+K)N(t)\log\left(\frac{N(t)}{d_1+K}\right)}\right.$$
$$\left.+\kappa c_\theta\right)+\frac{\max\{\alpha_k,\beta_k\}}{2}\left\|\Upsilon_k(t)\right\|_{U_t^{-1}}\left(\sqrt{2\lambda^2\log\left(\frac{\det(U_t)^{1/2}\det(\nu I)^{-1/2}}{\delta}\right)}+\right.$$
$$\left.c_\psi\sqrt{4(d_2+2K)N(t)\log\left(\frac{N(t)}{d_2+2K}\right)}+\eta c_\pi\right),$$

where $c_\sigma = \inf_{\theta,\pi,\Phi_k(s),\Upsilon_k(s)} \nabla\sigma\big(\Phi_k^T(s)\cdot\theta - f_k\big(\Upsilon_k(s);\pi\big)\big)$. Also, $N(t)$ is the number of unrealized

feedbacks when we are making a decision in time period t, which is sub-Gaussian and upper bounded by $2\mu_D + \tilde{\sigma}\sqrt{2\log\big(1/\delta\big)} + c$ with probability $1-\delta$, where $c = 2\,\tilde{\sigma}^2 \log\,(2\sigma_D^2+1)+1$ and $\tilde{\sigma} = \sigma_D\sqrt{p+2}$.

Proof. Proof of Proposition 1: Assume that k is the treatment selected by the algorithm for patient t. We decompose upper bounding the estimation loss (2.6) or $|\mathbf{V}_k(t) - \hat{\mathbf{V}}_k(t)|$, which is the difference between the true and estimated expected rewards of patient t under treatment k, into two parts:

$$\Big|\sigma\Big(\Phi_k^T(t)\cdot\theta - f_k\big(\Psi_k(t), y_k(t);\pi\big)\Big) - \sigma\Big(\Phi_k^T(t)\cdot\hat{\theta}(t) - f_k\big(\Psi_k(t), y_k(t);\hat{\pi}(t)\big)\Big)\Big| \tag{2.6}$$

$$\leq \underbrace{\Big|\sigma\Big(\Phi_k^T(t)\cdot\theta - f_k\big(\Psi_k(t), y_k(t);\pi\big)\Big) - \sigma\Big(\Phi_k^T(t)\cdot\hat{\theta}(t) - f_k\big(\Psi_k(t), y_k(t);\pi\big)\Big)\Big|}_{\text{Part I}} \tag{2.7}$$

$$+ \underbrace{\Big|\sigma\Big(\Phi_k^T(t)\cdot\hat{\theta}(t) - f_k\big(\Psi_k(t), y_k(t);\pi\big)\Big) - \sigma\Big(\Phi_k^T(t)\cdot\hat{\theta}(t) - f_k\big(\Psi_k(t), y_k(t);\ \hat{\pi}(t)\big)\Big)\Big|}_{\text{Part II}} \tag{2.8}$$

Part I (Estimation loss of parameter θ): We bound the loss (2.7) in two main steps: (1) deriving an online estimation $\hat{\theta}(t)$ for θ, and (2) developing a high-probability bound for the difference between the true expected reward $\sigma\big(\Phi_k^T(t)\cdot\theta - f_k\big(\Psi_k(t), y_k(t);\pi\big)\big)$ and the estimated expected reward $\sigma\big(\Phi_k^T(t)\cdot\hat{\theta}(t) - f_k\big(\Psi_k(t), y_k(t);\pi\big)\big)$ for patient t under the treatment k.

Step 1 (Online estimation of parameter θ): Recall that $R_k(t)$ is the uncertain binary reward related to the disease progression of patient t with the following expected value,

$$\mathbb{E}[R_k(t)] = \sigma\Big(\Phi_k^T(t)\cdot\theta - f_k\big(\Psi_k(t), y_k(t);\pi\big)\Big).$$

In our setting, patient feedback cannot be realized immediately after a decision is made for the patient. Indeed, true feedback is revealed with a *stochastic* delay. In the S-SGD Bandit algorithm, the model parameters get updated *on the fly*. This implies that we use the information of a patient to whom a treatment with a dosage was prescribed previously only if her/his feedback is realized up to the current time. Thus, the estimator of θ exploits only the realized context-feedback pairs in the history at time t. We denote $\mathcal{M}(t) = \underset{k\in\mathcal{K}}{\cup}\mathcal{M}_k(t)$ as the set of time-stamps with *realized* feedbacks by the end of time period $t-1$, where $\mathcal{M}_k(t) = \{s \mid s \leq t-1, s+D(s) \leq t-1, k(s)=k\}$. Moreover, we denote $\mathcal{N}(t) = \underset{k\in\mathcal{K}}{\cup}\mathcal{N}_k(t)$ as the set of time-stamps with *unrealized* feedbacks by the end of time period $t-1$, where $\mathcal{N}_k(t) = \{s \mid s \leq t-1, s+D(s) > t-1, k(s)=k\}$.

Let Θ be the set of admissible parameters θ. We define $\bar{\theta}(t)$ as the quasi-maximum likelihood estimator of the parameter $\theta \in \Theta$ at time period t. Given only the realized

feedbacks in the history at time period t, the regularized log-likelihood function $\mathcal{L}_t(\theta)$ can be calculated for $\kappa > 0$ as:

$$\mathcal{L}_t(\theta) = \sum_{s \in \mathcal{M}(t)} \Big[R_k(s) \log \sigma\big(\Phi_k^T(s) \cdot \theta - f_k\big(\Psi_k(s), y_k(s); \pi\big)\big) \\ + (1 - R_k(s)) \log \big(1 - \sigma\big(\Phi_k^T(s) \cdot \theta - f_k\big(\Psi_k(s), y_k(s); \pi\big)\big)\big) \Big] - \frac{\kappa}{2} \|\theta\|^2 .$$

Next, we need to find the maximum of $\mathcal{L}_t(\theta)$ so as to obtain the quasi-maximum likelihood estimator $\bar{\theta}(t)$. The derivative of $\mathcal{L}_t(\theta)$ with respect to the vector parameters θ as follows:

$$\nabla_\theta \mathcal{L}_t(\theta) = \sum_{s \in \mathcal{M}(t)} \Big(R_k(s) - \sigma\big(\Phi_k^T(s) \cdot \theta - f_k\big(\Psi_k(s), y_k(s); \pi\big)\big)\Big) \, \Phi_k(s) - \kappa\theta,$$

Therefore, $\bar{\theta}(t)$ is the unique solution of the estimating equation $\nabla_\theta \mathcal{L}_t(\theta) = 0$. However, this estimator might be outside of the admissible set Θ. To deal with this issue, we project $\bar{\theta}(t)$ back onto the set Θ, and derive the projected estimator $\hat{\theta}(t)$ as follows:

$$\hat{\theta}(t) = \arg\min_{\theta \in \Theta} \big\| h_t(\theta) - h_t(\bar{\theta}(t)) \big\|_{V_t^{-1}} ,$$

where $h_t(\theta) = \sum_{s=1}^{t-1} \sigma\big(\Phi_k^T(s) \cdot \theta - f_k\big(\Psi_k(s), y_k(s); \pi\big)\big) \, \Phi_k(s) + \kappa\theta$, and $V_t = \sum_{s=1}^{t-1} \Phi_k(s) \cdot \Phi_k^T(s) + \gamma I$ is the design matrix corresponding to the first $t-1$ time-steps of the observed features, where $\gamma = \kappa / c_\sigma \geq 1$ is a scalar constant.

Step 2 (High-probability bound for the estimation of θ): According to the *Lipschitz* property of the logistic function $\sigma(.)$, and the *mean-value theorem* for the vector-valued functions, Lemma II.4 (see Appendix C) yields the following bound for each time period t:

$$\Big| \sigma\Big(\Phi_k^T(t) \cdot \theta - f_k\big(\Psi_k(t), y_k(t); \pi\big)\Big) - \sigma\Big(\Phi_k^T(t) \cdot \hat{\theta}(t) - f_k\big(\Psi_k(t), y_k(t); \pi\big)\Big) \Big| \\ \leq \frac{1}{2c_\sigma} \|\Phi_k(t)\|_{V_t^{-1}} \big\| h_t(\theta) - h_t(\bar{\theta}(t)) \big\|_{V_t^{-1}} .$$

Next, we decompose the expression $\Big\| h_t(\hat{\theta}(t)) - h_t(\theta) \Big\|_{V_t^{-1}}$ on the RHS of above bound as

follows:

$$\big\|h_t(\bar{\theta}(t)) - h_t(\theta)\big\|_{V_t^{-1}} \leq \Big\| \sum_{s\in\mathcal{M}(t)} \Big(\sigma\big(\Phi_k^T(s)\cdot\bar{\theta}(t) - f_k\big(\Psi_k(s), y_k(s);\pi\big)\big) - \sigma\big(\Phi_k^T(s)\cdot\theta - f_k\big(\Psi_k(s), y_k(s);\pi\big)\big)\Big)\Phi_k(s) + \kappa(\bar{\theta}(t)-\theta)\Big\|_{V_t^{-1}} + \Big\| \sum_{s\in\mathcal{N}(t)} \Big(\sigma\big(\Phi_k^T(s)\cdot\bar{\theta}(t) - f_k\big(\Psi_k(s), y_k(s);\pi\big)\big) - \sigma\big(\Phi_k^T(s)\cdot\theta - f_k\big(\Psi_k(s), y_k(s);\pi\big)\big)\Big)\Phi_k(s)\Big\|_{V_t^{-1}},$$

where the inequality holds by triangle inequality. Below, we bound each term on the right-hand side of the above inequality, separately.

Section I: We first bound the first term on the right-hand side of the above inequality. From $\nabla_\theta \mathcal{L}_t(\theta) = 0$, we have the following over the set $\mathcal{M}(t)$ for each t:

$$\sum_{s\in\mathcal{M}(t)} \sigma\big(\Phi_k^T(s)\cdot\bar{\theta}(t) - f_k\big(\Psi_k(s), y_k(s);\pi\big)\big)\Phi_k(s) + \kappa\,\bar{\theta}(t) = \sum_{s\in\mathcal{M}(t)} R_k(s)\;\Psi_k(s).$$

Consequently, we can derive the following bound for Section I:

$$\begin{aligned}
&\Big\| \sum_{s\in\mathcal{M}(t)} \Big(\sigma\big(\Phi_k^T(s)\cdot\bar{\theta}(t) - f_k\big(\Psi_k(s), y_k(s);\pi\big)\big) - \\
&\sigma\big(\Phi_k^T(s)\cdot\theta - f_k\big(\Psi_k(s), y_k(s);\pi\big)\big)\Big)\Phi_k(s) + \kappa(\bar{\theta}(t)-\theta)\Big\|_{V_t^{-1}} \\
&= \left\| \sum_{s\in\mathcal{M}(t)} \Big(R_k(s) - \sigma\big(\Phi_k^T(s)\cdot\theta - f_k\big(\Psi_k(s), y_k(s);\pi\big)\big)\Big)\Phi_k(s) - \kappa\theta \right\|_{V_t^{-1}} \\
&\leq \left\| \sum_{s\in\mathcal{M}(t)} \xi_k(s)\;\Phi_k(s) \right\|_{V_t^{-1}} + \kappa\,\|\theta\|_{V_t^{-1}} \;\leq\; \left\| \sum_{s=1}^{t-1} \xi_k(s)\;\Phi_k(s) \right\|_{V_t^{-1}} + \kappa c_\theta, \qquad (2.9)
\end{aligned}$$

where the first equality holds since $\bar{\theta}(t)$ is the unique solution of $\nabla_\theta \mathcal{L}_t(\theta) = 0$. The first inequality is by triangle inequality and knowing that $\xi_k(t) = R_k(t) - \sigma\big(\Phi_k^T(t)\cdot\theta - f_k(\Psi_k(t), y_k(t);\pi)\big)$. The second inequality holds because $\|\theta\|^2_{V_t^{-1}} \leq \lambda^{-1}_{min}(V_t)\,\|\theta\|^2 \leq \|\theta\|^2$ and $\|\theta\| \leq c_\theta$.

Moreover, recall that the error term $\xi_k(t)$ is a conditionally 1-sub-Gaussian random variable, and also $V_t = \sum_{s=1}^{t-1} \Phi_k(s)\cdot\Phi_k^T(s) + \gamma I$, then $\sum_{s=1}^{t-1} \xi_k(s)\;\Phi_k(s)$ is a vector-valued *martingale* (see Definition 3 in Appendix C). Accordingly, we can show that this martingale stays close to zero with high probability. That is, for each time period t and $\delta > 0$, with

probability at least $1-\delta$, the following bound holds (see Theorem 1 in [1]):

$$\left\|\sum_{s=1}^{t-1}\xi(s)\;\Phi_k(s)\right\|_{V_t^{-1}} \leq \sqrt{2\log\left(\frac{\det(V_t)^{1/2}\det(\gamma I)^{-1/2}}{\delta}\right)}. \tag{2.10}$$

Section II: We have the following bound over the set $\mathcal{N}(t)$ for each time period t:

$$\begin{aligned}&\left\|\sum_{s\in\mathcal{N}(t)}\Big(\sigma\big(\Phi_k^T(s)\cdot\bar{\theta}(t)-f_k\big(\Psi_k(s),y_k(s);\pi\big)\big)-\sigma\big(\Phi_k^T(s)\cdot\theta-f_k\big(\Psi_k(s),y_k(s);\pi\big)\big)\Big)\Phi_k(s)\right\|_{V_t^{-1}}\\ &\leq\left\|\sum_{s\in\mathcal{N}(t)}\Phi_k(s)\right\|_{V_t^{-1}}\leq\sum_{s\in\mathcal{N}(t)}\|\Phi_k(s)\|_{V_t^{-1}}\leq\sqrt{2(d_1+K)N(t)\log\left(\frac{N(t)}{d_1+K}\right)}.\end{aligned} \tag{2.11}$$

The first inequality holds because the logistic function implies $\big|\sigma\big(x\big)-\sigma\big(y\big)\big|\leq 1$ for each x and y, and the second one holds by triangle inequality. The third inequality is obtained using Lemma II.5 (see Appendix C), where $N(t)=|\mathcal{N}(t)|$. Note that $N(t)=\sum_{s=1}^{t-1}\mathbb{1}\big\{s+D(s)\geq t\big\}$ is the random number of *unrealized* feedbacks when we are making a decision in time period t. In Lemma II.6 (see Appendix C), by building a sequence of stopping times for the number of realized feedbacks, we characterize the tail behavior of N_t, so that $N(t)$ is sub-Gaussian and $N(t)\leq 2\mu_D+\tilde{\sigma}\sqrt{2\log\big(1/\delta\big)}+c$ with probability $1-\delta$, where $c=2\,\tilde{\sigma}^2\log\,(2\sigma_D^2+1)+1$ and $\tilde{\sigma}=\sigma_D\sqrt{p+2}$.

Putting the high-probability bounds developed in Sections I and II together, Part I in (2.7) can be bounded with probability at least $1-\delta$.

Part II (Estimation loss of parameter π): From §2.2, recall that $\Pi_k(t)$ is the uncertain treatment-dosage sub-outcome with the following expected value,

$$\mathbb{E}\big[\Pi_k(t)\big]=\Psi_k^T(t)\cdot\omega+y_k^T(t)\cdot\tau.$$

Note that $\pi=(\omega^T,\tau^T)^T$, and let $\Upsilon_k(t)=(\Psi_k^T(t),y_k^T(t))^T$ for the ease of notation. We define $\bar{\pi}(t)$ as the maximum-likelihood estimator of π at time period t with the regularization parameter $\eta>0$. Since the linear model can be viewed as a generalized linear model with the above linear link function, this estimator is the unique solution of $\nabla_\pi\mathcal{U}_t(\pi)=0$, where:

$$\nabla_\pi\mathcal{U}_t(\pi)=\sum_{s\in\mathcal{M}(t)}\Big(\Pi_k(s)-\Upsilon_k^T(s)\cdot\pi\Big)\;\Upsilon_k(s)-\eta\pi.$$

Let Λ be the set of admissible parameters π. To ensure that the estimated parameter $\bar{\pi}(t)$ belongs to this admissible set, we define the projected estimator $\hat{\pi}(t)$, which projects $\bar{\pi}(t)$

back onto Λ as:

$$\hat{\pi}(t) = \arg\min_{\pi \in \Lambda} \| g_t(\pi) - g_t(\bar{\pi}(t)) \|_{U_t^{-1}},$$

where $g_t(\pi) = \sum_{s=1}^{t-1} \left(\Upsilon_k^T(s) \cdot \pi\right) \Upsilon_k(s) + \eta\pi$, and $U_t = \sum_{s=1}^{t-1} \Upsilon_k(s) \cdot \Upsilon_k^T(s) + \nu I$ is the design matrix corresponding to the first $t-1$ time-steps of the observed features.

Next, we establish a high-probability bound for the loss due to the estimation of π. First, Lemma II.4 (see Appendix C) for the *linear* vector-valued function $g(.)$ derives the following bound:

$$\left| \Upsilon_k^T(t) \cdot \pi - \Upsilon_k^T(t) \cdot \hat{\pi}(t) \right| \leq 2 \, \| \Upsilon_k(t) \|_{U_t^{-1}} \, \| g_t(\pi) - g_t(\bar{\pi}(t)) \|_{U_t^{-1}}. \tag{2.12}$$

Similar to Part I, we decompose the term $\| g_t(\pi) - g_t(\bar{\pi}(t)) \|_{U_t^{-1}}$ on the RHS of the above as follows:

$$\| g_t(\pi) - g_t(\bar{\pi}(t)) \|_{U_t^{-1}} \leq \underbrace{\left\| \sum_{s \in \mathcal{M}(t)} \Big(\Upsilon_k^T(s) \cdot \bar{\pi}(t) - \Upsilon_k^T(s) \cdot \pi \Big) \Upsilon_k(s) + \eta \, (\bar{\pi}(t) - \pi) \right\|_{U_t^{-1}}}_{\text{Section I}}$$
$$+ \underbrace{\left\| \sum_{s \in \mathcal{N}(t)} \Big(\Upsilon_k^T(s) \cdot \bar{\pi}(t) - \Upsilon_k^T(s) \cdot \pi \Big) \Upsilon_k^T(s) \right\|_{U_t^{-1}}}_{\text{Section II}},$$

where the inequality holds by triangle inequality. Below, we bound each section separately.
Section I: We establish the following over the set $\mathcal{M}(t)$ for time period t with probability $1-\delta$:

$$\left\| \sum_{s \in \mathcal{M}(t)} \Big(\Upsilon_k^T(s) \cdot \bar{\pi}(t) - \Upsilon_k^T(s) \cdot \pi \Big) \Upsilon_k(s) + \eta \, (\bar{\pi}(t) - \pi) \right\|_{U_t^{-1}} \leq \left\| \sum_{s \in \mathcal{M}(t)} \zeta_k(s) \, \Upsilon_k(s) \right\|_{U_t^{-1}}$$
$$+ \eta \, \|\pi\|_{U_t^{-1}} \leq \left\| \sum_{s=1}^{t-1} \zeta_k(s) \, \Upsilon_k(s) \right\|_{U_t^{-1}} + \eta \, \|\pi\| \leq \sqrt{2\lambda^2 \log\left(\frac{\det(U_t)^{1/2} \det(\nu I)^{-1/2}}{\delta} \right)} + \eta \, c_\pi.$$

The first inequality holds since $\bar{\pi}(t)$ is the unique solution of $\nabla_\pi \mathcal{U}_t(\pi) = 0$ and knowing that $\zeta_k(t) = \Pi_k(t) - \Upsilon_k^T(t) \cdot \pi$. The second inequality holds because $\|\pi\|_{U_t^{-1}}^2 \leq \lambda_{min}^{-1}(U_t) \, \|\pi\|^2 \leq \|\pi\|^2$. Recall that the error term $\zeta_k(t)$ is a conditionally λ-sub-Gaussian random variable, then $\sum_{s=1}^{t-1} \zeta_k(s) \, \Upsilon_k(s)$ is a vector-valued martingale. Consequently, the last inequality holds with probability at least $1-\delta$ for each time period t and $\delta > 0$ (see Theorem 1 in [1]).

Section II: We develop the following over the set $\mathcal{N}(t)$ for time period t:

$$\Big\| \sum_{s\in\mathcal{N}(t)} \Big(\Upsilon_k^T(s)\cdot\bar{\pi}(t) - \Upsilon_k^T(s)\cdot\pi\Big)\Upsilon_k(s)\Big\|_{U_t^{-1}} \leq 2c_\psi \sum_{s\in\mathcal{N}(t)} \|\Upsilon_k(s)\|_{U_t^{-1}}$$
$$\leq 2c_\psi\sqrt{2(d_2+2K)N(t)\log\left(\frac{N(t)}{d_2+2K}\right)}.$$

The first inequality holds by triangle inequality and $\big|\Pi_k(s)\big| \leq c_\psi$. The second inequality is established using Lemma II.5 (see Appendix C).

Moreover, note that the dosage penalty function $f_k\big(\Psi_k(t), y_k(t); \pi\big)$ is Lipschitz with constant $\max\{\alpha_k, \beta_k\}$ due to reverse triangle inequality $|\,\|x\| - \|y\|\,| \leq \|x-y\|$ for $x, y \in \mathbb{R}^n$. Leveraging the Lipschitz property of the logistic function $\sigma(.)$, the following confidence bound can be established using the upper bound derived for (2.12) in Sections I and II with probability at least $1-\delta$:

$$\Big|\sigma\Big(\Phi_k^T(t)\cdot\hat{\theta}(t) - f_k\big(\Psi_k(t), y_k(t);\pi\big)\Big) - \sigma\Big(\Phi_k^T(t)\cdot\hat{\theta}(t) - f_k\big(\Psi_k(t), y_k(t);\hat{\pi}(t)\big)\Big)\Big| \tag{2.13}$$
$$\leq \frac{1}{4}\Big|f_k\big(\Psi_k(t), y_k(t);\pi\big) - f_k\big(\Psi_k(t), y_k(t);\hat{\pi}(t)\big)\Big| \leq \frac{\max\{\alpha_k,\beta_k\}}{4}\Big|\Upsilon_k^T(t)\cdot\pi - \Upsilon_k^T(t)\cdot\hat{\pi}(t)\Big|$$
$$\leq \frac{\max\{\alpha_k,\beta_k\}}{2}\|\Upsilon_k(t)\|_{U_t^{-1}}\left(\sqrt{2\lambda^2\log\left(\frac{\det(U_t)^{1/2}\det(\nu I)^{-1/2}}{\delta}\right)} + \right.$$
$$\left. c_\psi\sqrt{4(d_2+2K)N(t)\log\left(\frac{N(t)}{d_2+2K}\right)} + \eta c_\pi\right).$$

Therefore, plugging all the above-derived results of Part I established in (2.9)-(2.11) as well as the result of Part II established in (2.13), into inequality (2.6) completes the proof. □

Next, we bound the contextual bandit loss $\mathbb{E}\big[\sum_{t=1}^T \big(\mathbf{V}_{k^*}^*(t) - \mathbf{V}_k^*(t)\big)\big]$ that is incurred due to learning the optimal treatment decision k^* using the result of Proposition 1.

Proposition 2 (**Contextual Bandit Loss**). For any $\delta > 0$, the following bound on the total loss over T time periods corresponding to the difference between the true expected reward of the optimal treatment k^* with the optimal dosage $y_{k^*}^*$, and the true expected reward of a treatment k chosen by the proposed algorithm with the optimal dosage y_k^*, holds with probability at least $1-2\delta$:

$$\mathbb{E}\Big[\sum_{t=1}^T \big(\mathbf{V}_{k^*}^*(t) - \mathbf{V}_k^*(t)\big)\Big] \leq L(T,\delta) + Q(T,\delta) + 2T\delta, \tag{2.14}$$

where $L(T,\delta)$ and $Q(T,\delta)$ are defined respectively as

$$L(T,\delta) = \frac{1}{c_\sigma}\sqrt{2T(d_1+K)\log\left(1+\frac{T}{\gamma(d_1+K)}\right)}$$
$$\left(\sqrt{(d_1+K)\log\left(1+\frac{T}{\gamma(d_1+K)}\right)+\log\left(\frac{1}{\delta^2}\right)}+\sqrt{2(d_1+K)N_{\max}\log\left(\frac{N_{\max}}{d_1+K}\right)}+\kappa c_\theta\right),$$

$$Q(T,\delta) = \max\{\alpha_k,\beta_k\}\sqrt{2T(d_2+2K)\log\left(1+\frac{T}{\nu(d_2+2K)}\right)}$$
$$\left(\lambda\sqrt{(d_2+2K)\log\left(1+\frac{T}{\nu(d_2+2K)}\right)+\log\left(\frac{1}{\delta^2}\right)}+\right.$$
$$\left.c_\psi\sqrt{4(d_2+2K)N_{\max}\log\left(\frac{N_{\max}}{d_2+2K}\right)}+\eta c_\pi\right),$$

and $N_{\max} = \max_{1\le t\le T} N(t)$ that is defined in Theorem II.1.

Proof. Proof of Proposition 2: To bound the total contextual bandit loss, we switch from an upper-confidence bound argument to a TS-based one. Proposition 1 builds a confidence set Ω_t defined in (2.5), which contains the true parameters (θ,π) with probability at least $1-2\delta$ in each time period t. We define the following upper and lower bounds:

$$UB_k(t) := \min\left\{1, \max_{(\theta,\pi)\in\Omega_t} \sigma\big(\Phi_k^T(t)\cdot\theta - f_k\big(\Psi_k(t), y_k^*;\pi\big)\big)\right\}, \tag{2.15}$$

$$LB_k(t) := \max\left\{0, \min_{(\theta,\pi)\in\Omega_t} \sigma\big(\Phi_k^T(t)\cdot\theta - f_k\big(\Psi_k(t), y_k^*;\pi\big)\big)\right\}. \tag{2.16}$$

These quantities indicate the largest and smallest possible values for the expected reward based on the history $\mathcal{H}_t$, respectively. They are sequences of real-valued functions of $\mathcal{H}_t$, feature vectors $\Phi_k(t)$ and $\Psi_k(t)$, and the optimal dosage vector y_k^*. The contextual bandit

loss is bounded by:

$$
\begin{aligned}
&\mathbb{E}\Big[\sum_{t=1}^{T}\big(\mathbf{V}^*_{k^*}(t)-\mathbf{V}^*_k(t)\big)\Big]\\
&=\sum_{t=1}^{T}\mathbb{E}\Big[\Big(\sigma\big(\Phi^T_{k^*}(t)\cdot\theta-f_{k^*}\big(\Psi_{k^*}(t),y^*_{k^*};\pi\big)\big)-\sigma\big(\Phi^T_k(t)\cdot\theta-f_k\big(\Psi_k(t),y^*_k;\pi\big)\big)\Big)\Big]\\
&=\sum_{t=1}^{T}\mathbb{E}\Big[\mathbb{E}\Big[\Big(\sigma\big(\Phi^T_{k^*}(t)\cdot\theta-f_{k^*}\big(\Psi_{k^*}(t),y^*_{k^*};\pi\big)\big)-\sigma\big(\Phi^T_k(t)\cdot\theta-f_k\big(\Psi_k(t),y^*_k;\pi\big)\big)\Big)\Big|\mathcal{H}_t\Big]\Big]\\
&=\sum_{t=1}^{T}\mathbb{E}\Big[\mathbb{E}\Big[\Big(UB_k(t)-UB_k(t)+\sigma\big(\Phi^T_k(t)\cdot\theta-f_{k^*}\big(\Psi_{k^*}(t),y^*_{k^*};\pi\big)\big)\\
&-\sigma\big(\Phi^T_k(t)\cdot\theta-f_k\big(\Psi_k(t),y^*_k;\pi\big)\big)\Big)\Big|\mathcal{H}_t\Big]\Big]\\
&=\sum_{t=1}^{T}\mathbb{E}\Big[\mathbb{E}\Big[\Big(UB_k(t)-\sigma\big(\Phi^T_k(t)\cdot\theta-f_k\big(\Psi_k(t),y^*_k;\pi\big)\big)\Big)\Big|\mathcal{H}_t\Big]\Big]\\
&+\sum_{t=1}^{T}\mathbb{E}\Big[\mathbb{E}\Big[\Big(\sigma\big(\Phi^T_{k^*}(t)\cdot\theta-f_{k^*}\big(\Psi_{k^*}(t),y^*_{k^*};\pi\big)\big)-UB_{k^*}(t)\Big)\Big|\mathcal{H}_t\Big]\Big]\\
&=\underbrace{\sum_{t=1}^{T}\mathbb{E}\Big[\Big(UB_k(t)-\sigma\big(\Phi^T_k(t)\cdot\theta-f_k\big(\Psi_k(t),y^*_k;\pi\big)\big)\Big)\Big]}_{\text{Part I}}\\
&+\underbrace{\sum_{t=1}^{T}\mathbb{E}\Big[\Big(\sigma\big(\Phi^T_{k^*}(t)\cdot\theta-f_{k^*}\big(\Psi_{k^*}(t),y^*_{k^*};\pi\big)\big)-UB_{k^*}(t)\Big)\Big]}_{\text{Part II}},
\end{aligned}
$$

where in the fourth equality, we use the fact that conditional on $\mathcal{H}_t$, the optimal treatment k^* and the treatment k selected by the algorithm are identically distributed; thus, $\mathbb{E}\big(UB_k(t)|\mathcal{H}_t\big)=\mathbb{E}\big(UB_{k^*}(t)|\mathcal{H}_t\big)$. Note that to derive the above decomposition, we leverage the connection between TS based algorithms and UCB based algorithms ([144]). We then separately bound each part of the last line of the above equation as follows.

Part I: Using the definition of the upper and lower bounds in (2.15) and (2.16), we have:

$$
\sum_{t=1}^{T}\mathbb{E}\Big[\Big(UB_k(t)-\sigma\big(\Phi^T_k(t)\cdot\theta-f_k\big(\Psi_k(t),y^*_k;\pi\big)\big)\Big)\Big]\le\sum_{t=1}^{T}\mathbb{E}\Big[\Big(UB_k(t)-LB_k(t)\Big)\Big]. \quad (2.17)
$$

From Proposition 1, recall that for any t and $\delta > 0$, the following bound holds:

$$\left|\sigma\Big(\Phi_k^T(t)\cdot\theta - f_k\big(\Psi_k(t), y_k^*; \pi\big)\Big) - \sigma\Big(\Phi_k^T(t)\cdot\hat{\theta}(t) - f_k\big(\Psi_k(t), y_k^*; \hat{\pi}(t)\big)\Big)\right| \leq \Lambda_t^k(\delta), \quad (2.18)$$

with probability at least $1-2\delta$, where

$$\Lambda_t^k(\delta) = \frac{1}{2c_\sigma}\|\Phi_k(t)\|_{V_t^{-1}}\left(\sqrt{2\log\left(\frac{\det(V_t)^{1/2}\det(\gamma I)^{-1/2}}{\delta}\right)}\right.$$
$$\left.+\sqrt{2(d_1+K)N(t)\log\left(\frac{N(t)}{d_1+K}\right)}+\kappa c_\theta\right)+$$
$$\frac{\max\{\alpha_k,\beta_k\}}{2}\|\Upsilon_k(t)\|_{U_t^{-1}}\left(\sqrt{2\lambda^2\log\left(\frac{\det(U_t)^{1/2}\det(\nu I)^{-1/2}}{\delta}\right)}\right.$$
$$\left.+c_\psi\sqrt{4(d_2+2K)N(t)\log\left(\frac{N(t)}{d_2+2K}\right)}+\eta c_\pi\right).$$

Furthermore, we define the sequences $\overline{UB}_k(t)$ and $\overline{LB}_k(t)$ as follows:

$$\overline{UB}_k(t) := \sigma\Big(\Phi_k^T(t)\cdot\hat{\theta}(t) - f_k\big(\Psi_k(t), y_k^*; \hat{\pi}(t)\big)\Big) + \Lambda_t^k(\delta), \quad (2.19)$$

$$\overline{LB}_k(t) := \sigma\Big(\Phi_k^T(t)\cdot\hat{\theta}(t) - f_k\big(\Psi_k(t), y_k^*; \hat{\pi}(t)\big)\Big) - \Lambda_t^k(\delta). \quad (2.20)$$

Since $UB_k(t) \leq \overline{UB}_k(t)$ and $LB_k(t) \geq \overline{LB}_k(t)$, $UB_k(t) - LB_k(t) \leq \overline{UB}_k(t) - \overline{LB}_k(t)$. Taking the expectation on both sides of this inequality and summing over all periods yield the following:

$$\sum_{t=1}^{T}\mathbb{E}\Big[\Big(UB_k(t) - LB_k(t)\Big)\Big] \leq 2\sum_{t=1}^{T}\mathbb{E}[\Lambda_t^k(\delta)]. \quad (2.21)$$

Using a similar argument that we made in Part I of the proof of Corollary II.2 (see Appendix B), the summation of expectations $\mathbb{E}[\Lambda_t^k(\delta)]$ over T time periods on the RHS of

(2.21) is bounded as:

$$\sum_{t=1}^{T}\mathbb{E}[\Lambda_t^k(\delta)] \leq$$
$$\frac{1}{2c_\sigma}\sqrt{2T(d_1+K)\log\left(1+\frac{T}{\gamma(d_1+K)}\right)}\left(\sqrt{(d_1+K)\log\left(1+\frac{T}{\gamma(d_1+K)}\right)+\log\left(\frac{1}{\delta^2}\right)}+\right.$$
$$\left.+\sqrt{2(d_1+K)N_{\max}\log\left(\frac{N_{\max}}{d_1+K}\right)}+\kappa c_\theta\right)+\frac{\max\{\alpha_k,\beta_k\}}{2}$$
$$\sqrt{2T(d_2+2K)\log\left(1+\frac{T}{\nu(d_2+2K)}\right)}\left(\lambda\sqrt{(d_2+2K)\log\left(1+\frac{T}{\nu(d_2+2K)}\right)+\log\left(\frac{1}{\delta^2}\right)}\right.$$
$$\left.+c_\psi\sqrt{4(d_2+2K)N_{\max}\log\left(\frac{N_{\max}}{d_2+2K}\right)}+\eta c_\pi\right).$$

Therefore, replacing the above bound into (2.21) and correspondingly (2.18), Section I is then upper bounded with probability at least $1-2\delta$.

Part II: Since $\sigma\big(\Phi_{k^*}^T(t)\cdot\theta - f_{k^*}\big(\Psi_{k^*}(t),y_{k^*}^*;\pi\big)\big)\in[0,1]$ and also $UB_{k^*}(t)\in[0,1]$, we have:

$$\sum_{t=1}^{T}\Big(\sigma\big(\Phi_{k^*}^T(t)\cdot\theta - f_{k^*}\big(\Psi_{k^*}(t),y_{k^*}^*;\pi\big)\big)-UB_{k^*}(t)\Big)$$
$$\leq\sum_{t=1}^{T}\mathbb{1}\Big(\sigma\big(\Phi_{k^*}^T(t)\cdot\theta - f_{k^*}\big(\Psi_{k^*}(t),y_{k^*}^*;\pi\big)\big)>UB_{k^*}(t)\Big).$$

Taking the expectation from both sides results in the following:

$$\sum_{t=1}^{T}\mathbb{E}\Big[\big(\sigma\big(\Phi_{k^*}^T(t)\cdot\theta - f_{k^*}(\Psi_{k^*}(t),y_{k^*}^*;\pi)\big)-UB_{k^*}(t)\big)\Big]$$
$$\leq\sum_{t=1}^{T}\mathbb{P}\Big(\sigma\big(\Phi_{k^*}^T(t)\cdot\theta - f_{k^*}(\Psi_{k^*}(t),y_{k^*}^*;\pi)\big)>UB_{k^*}(t)\Big).$$

Next, according to the definitions of $\overline{UB}_k(t)$ and $\overline{LB}_k(t)$, and also using the high-probability bound obtained in Proposition 1, we have the following high-probability bound:

$$\mathbb{P}\Big(\overline{LB}_{k^*}(t)\leq\sigma\left(\Phi_{k^*}^T(t)\cdot\theta - f_{k^*}\left(\Psi_{k^*}(t),y_{k^*}^*;\pi\right)\right)\leq\overline{UB}_{k^*}(t)\Big)\geq 1-2\delta.$$

Accordingly, we have the following result:

$$\sum_{t=1}^{T} \mathbb{P}\Big(\sigma\big(\Phi_{k^*}^T(t)\cdot\theta - f_{k^*}\big(\Psi_{k^*}(t), y_{k^*}^*; \pi\big)\big) > UB_{k^*}(t)\Big)$$
$$= \sum_{t=1}^{T} \mathbb{P}\Big(\sigma\big(\Phi_{k^*}^T(t)\cdot\theta - f_{k^*}\big(\Psi_{k^*}(t), y_{k^*}^*; \pi\big)\big) > \overline{UB}_{k^*}(t)\Big) \leq 2T\delta.$$

In the above equality, LHS could not be less than RHS since $UB_{k^*}(t) \leq \overline{UB}_{k^*}(t)$. The only possible case for LHS to be greater than RHS is when $UB_{k^*}(t) < \overline{UB}_{k^*}(t) < 1$, which is not possible based on the definition of $UB_{k^*}(t)$. Thus, the equality holds since LHS and RHS are always equal.

Finally, putting both bounds developed in Sections I and II together, we establish the contextual learning loss bound, which completes the proof. □

Remark. Note that the contextual learning loss in Proposition 2 is obtained under the assumption of *delayed* feedbacks. If there were *no* delay in observing the patient's feedbacks, then we would not have those terms incorporating $N_{\max}$ on the RHS of this bound. Therefore, the terms incorporating $N_{\max}$ represent the effect of delayed feedback on this contextual learning loss.

2.4.5 Theoretical Results on Online Sub-Gaussian Stochastic Sub-Gradient Descent

Although we know the form of the penalty function in (2.2), there is an unknown parameter π, which hinders us from finding the optimal dosage of a given treatment. Further, there is a stochastic delay in observing the noisy gradient of this function. We develop a sub-Gaussian Stochastic Sub-Gradient Descent (S-SGD) procedure under stochastic delay to find the optimal dosage. We prove that the loss due to the sub-optimality of dosage (*S-SGD sub-optimality loss*) is bounded using a new high-probability regret bound. Since this technical result is of independent interest beyond the scope of this paper, we state and prove it with *general sub-differentiable and convex* functions.

Proposition 3 (**High-Probability Regret Bound for Online Sub-Gaussian S-SGD**). Let $\{y_i\}_{i=1}^n$ be a sequence obtained by the projected online S-SGD algorithm under stochastic delay with respect to sub-differentiable and convex functions $f(.)$ with a domain $\mathcal{K}$, i.e.,

$$y_1 \in \mathcal{K},\ y_{i+1} = \mathbf{Proj}_{\mathcal{K}}\Big(y_i - \eta \sum_{s\in\mathcal{S}_i} \widetilde{\nabla} f_s\Big),\ \ \forall i = 1, ..., n-1,$$

where $\mathcal{S}_i = \{s \in [n] \mid s + D(s) - 1 = i\}$ is the set of iterations whose feedback appears at the end of iteration i, $D(s)$ is the stochastic non-negative delay, $\widetilde{\nabla} f_s \in \partial f_s(y_s)$ is a stochastic sub-gradient of f_s at y_s, and η is the step size at each iteration. We make the following assumptions:

1. Diameter of the domain $\mathcal{K}$ is bounded by a constant G, i.e., $\sup_{y_1, y_2 \in \mathcal{K}} \|y_1 - y_2\| \leq G$.
2. For each $i = 1, ..., n-1$, conditional on y_i, the stochastic sub-gradient $\widetilde{\nabla} f_i \in \partial f_i(y_i)$ is a ρ-sub-Gaussian random vector with the second moment bounded by a positive constant ξ^2.
3. The stochastic delay $D(s)$ satisfies the regularity condition (2.4), and $D = \sum_{i=1}^{n} D(i)$.

If we choose $\eta = \frac{G}{\xi\sqrt{n+D}}$, the following bound holds with probability at least $1 - 2\delta$:

$$\frac{1}{n}\sum_{i=1}^{n}\Big(f_i(y_i) - f_i(y^*)\Big) \leq \sqrt{\frac{2G^2\rho\log(1/\delta)}{n}} + \left(\sqrt{1 + \mu_D + \sqrt[p+1]{\sigma_D^2 \log(n/\delta)}}\right)\frac{3G\xi}{\sqrt{n}}, \quad (2.22)$$

where $y^* = \arg\min_{y \in \mathcal{K}} \sum_{i=1}^{n} f_i(y)$.

Remark. The choice of η requires prior knowledge of the total delay D; however, one can calculate D on the fly. That is, if there exist τ unrealized feedbacks at one iteration, then D increases by exactly τ. Obviously, we have $\tau \leq n$ and $n \leq D$, so D doubles at most. Thus, at the cost of slightly worse constants, we can use the *doubling trick* (i.e., setting a budget on the unknown quantity and restarting the algorithm with a double budget when the budget is depleted) to dynamically adjust η as D increases (see [45] for details).

Proof. Proof of Proposition 3: First, since there is a delay $D(s)$ for each iteration s, the feedback from iteration s is revealed at the end of iteration $s + D(s) - 1$, and so can be exploited in iteration $s + D(s)$. By convexity of $f(.)$, we have the following for each *realized* feedback:

$$f_i(y^*) \geq f_i(y_i) + \widehat{\nabla} f_i(y_i)\,(y^* - y_i), \quad \forall i = 1, ..., n,$$

where $\widehat{\nabla} f_i(y_i) \in \partial f_i(y_i) = \big\{g : f_i(u) \geq f_i(y_i) + g^T(u - y_i), \forall u\big\}$, and $\partial f_i(y_i)$ is the set of all sub-gradients of $f_i(.)$ at y_i. Using the above property, we establish the following

decomposition:

$$\frac{1}{n}\sum_{i=1}^{n}\Big(f_i(y_i)-f_i(y^*)\Big) \le \frac{1}{n}\sum_{i=1}^{n}\langle\,\widehat{\nabla} f_i(y_i), y_i-y^*\rangle$$
$$= \underbrace{\frac{1}{n}\sum_{i=1}^{n}\langle\,\widetilde{\nabla} f_i, y_i-y^*\rangle}_{\text{Part I}} + \underbrace{\frac{1}{n}\sum_{i=1}^{n}\langle\,\widehat{\nabla} f_i(y_i)-\widetilde{\nabla} f_i, y_i-y^*\rangle}_{\text{Part II}}. \quad (2.23)$$

where $\widetilde{\nabla} f_i$ is a stochastic sub-gradient of f_i at y_i. We next bound each part on RHS of (2.23), below.

Part I: First, rewrite that $y_{i+1} = \mathbf{Proj}_{\mathcal{K}}\big(\widetilde{y}_{i+1}\big)$, where $\widetilde{y}_{i+1} = y_i - \eta\sum_{s\in\mathcal{S}_i}\widetilde{\nabla} f_s$, for ease of notation. We break the sum of sub-gradients used in one iteration and consider them one by one. That is, for each $s\in\mathcal{S}_i$, we define $\mathcal{S}_{i,s}=\{q\in\mathcal{S}_i \mid q<s\}$ and $y_{i,s} = y_i - \eta\sum_{q\in\mathcal{S}_{i,s}}\widetilde{\nabla} f_q$. Let $s_{\max}$ be the last or maximum index in $\mathcal{S}_i\neq\emptyset$, by using the properties of projection operator, we then have:

$$\|y_{i+1}-y^*\|^2 \le \|\widetilde{y}_{i+1}-y^*\|^2 = \Big\|y_{i,s_{\max}} - \eta\,\widetilde{\nabla}_{s_{\max}} f - y^*\Big\|^2$$
$$= \|y_{i,s_{\max}}-y^*\|^2 + \eta^2\,\Big\|\widetilde{\nabla} f_{s_{\max}}\Big\|^2 - 2\eta\,\langle\,\widetilde{\nabla} f_{s_{\max}},\; y_{i,s_{\max}}-y^*\rangle.$$

If we keep expanding the term $\|y_{i,s_{\max}}-y^*\|^2$ in the last expression in a similar fashion, we will have the following for each iteration i:

$$\|y_{i+1}-y^*\|^2 \le \|y_i-y^*\|^2 + \eta^2\sum_{s\in\mathcal{S}_i}\Big\|\widetilde{\nabla} f_s\Big\|^2 - 2\eta\sum_{s\in\mathcal{S}_i}\langle\,\widetilde{\nabla} f_s,\; y_{i,s}-y^*\rangle,$$

which implies that

$$\sum_{s\in\mathcal{S}_i}\langle\,\widetilde{\nabla} f_s,\; y_{i,s}-y^*\rangle \le \frac{1}{2}\left(\frac{\|y_i-y^*\|^2-\|y_{i+1}-y^*\|^2}{\eta} + \eta\sum_{s\in\mathcal{S}_i}\Big\|\widetilde{\nabla} f_s\Big\|^2\right).$$

We then decompose Part I in (2.16) into two parts, and apply the above inequality to

establish:

$$\sum_{i=1}^{n}\langle\,\widetilde{\nabla} f_i, y_i - y^*\rangle = \sum_{i=1}^{n}\sum_{s\in\mathcal{S}_i}\langle\,\widetilde{\nabla} f_s, y_s - y^*\rangle = \sum_{i=1}^{n}\sum_{s\in\mathcal{S}_i}\Big(\langle\,\widetilde{\nabla} f_s, y_{i,s} - y^*\rangle + \langle\,\widetilde{\nabla} f_s, y_s - y_{i,s}\rangle\Big)$$
$$\leq \frac{1}{2}\sum_{i=1}^{n}\left(\frac{\|y_i - y^*\|^2 - \|y_{i+1} - y^*\|^2}{\eta} + \eta\sum_{s\in\mathcal{S}_i}\left\|\widetilde{\nabla} f_s\right\|^2 + 2\sum_{s\in\mathcal{S}_i}\langle\,\widetilde{\nabla} f_s, y_s - y_{i,s}\rangle\right)$$
$$\leq \frac{1}{2}\sum_{i=1}^{n}\left(\frac{\|y_i - y^*\|^2 - \|y_{i+1} - y^*\|^2}{\eta} + \eta\,\xi^2\,|\mathcal{S}_i| + 2\sum_{s\in\mathcal{S}_i}\langle\,\widetilde{\nabla} f_s, y_s - y_{i,s}\rangle\right)$$
$$\leq \left(\frac{G^2}{2\eta} + \frac{\eta}{2}n\xi^2\right) + \sum_{i=1}^{n}\sum_{s\in\mathcal{S}_i}\langle\,\widetilde{\nabla} f_s, y_s - y_{i,s}\rangle$$
$$\leq\ \xi G\sqrt{n+D} + \xi\sum_{i=1}^{n}\sum_{s\in\mathcal{S}_i}\|y_s - y_{i,s}\|\,. \tag{2.24}$$

The second inequality is by $\left\|\widetilde{\nabla} f_s\right\| \leq \xi$. To get the last inequality, the step size $\eta = \frac{G}{\xi\sqrt{n+D}}$ is used, and the *delay* expression $\sum_{i=1}^{n}\sum_{s\in\mathcal{S}_i}\langle\,\widetilde{\nabla} f_s, y_s - y_{i,s}\rangle$ is bounded by Cauchy-Schwartz inequality.

In (2.24), $\|y_s - y_{i,s}\|$ is the distance between the point y_s (where the feedback $\widetilde{\nabla}_s f$ is generated in iteration s) and the point $y_{i,s}$ (where this feedback is realized in iteration i and used). This distance is about the sum of feedbacks realized and used in between, and the number of such in-between realized feedbacks is closely related to the total delay. To see this, using triangle inequality and the property of projection, we first have the following:

$$\|y_{i,s} - y_s\| \leq \|y_{i,s} - y_i\| + \|y_i - y_s\| \leq \|y_{i,s} - y_i\| + \|\widetilde{y}_i - y_s\|\,.$$

Similarly, if we keep expanding the second term as $\|\widetilde{y}_i - y_s\| \leq \|\widetilde{y}_i - y_{i-1}\| + \|\widetilde{y}_{i-1} - y_s\|$, and take the sum over all iterations, we will have the following bound:

$$\sum_{i=1}^{n}\sum_{s\in\mathcal{S}_i}\|y_{i,s} - y_s\| \leq \sum_{i=1}^{n}\sum_{s\in\mathcal{S}_i}\left(\sum_{t=s}^{i-1}\|\widetilde{y}_{t+1} - y_t\| + \|y_{i,s} - y_i\|\right)$$
$$\leq \sum_{i=1}^{n}\sum_{s\in\mathcal{S}_i}\left(\eta\sum_{t=s}^{i-1}\sum_{p\in\mathcal{S}_t}\left\|\widetilde{\nabla} f_p\right\| + \eta\sum_{q\in\mathcal{S}_{i,s}}\left\|\widetilde{\nabla} f_q\right\|\right)$$
$$\leq\ \xi\,\eta\sum_{i=1}^{n}\sum_{s\in\mathcal{S}_i}\left(\sum_{t=s}^{i-1}|\mathcal{S}_t| + |\mathcal{S}_{i,s}|\right)\ \leq\ 2\,\xi\,\eta\sum_{i=1}^{n}D(i)\ =\ 2\,\xi\,\eta\,D. \tag{2.25}$$

The second inequality is established by $\widetilde{y}_{i+1} = y_i - \eta_i\sum_{s\in\mathcal{S}_i}\widetilde{\nabla} f_s$ and $y_{i,s} = y_i - \eta_i\sum_{q\in\mathcal{S}_{i,s}}\widetilde{\nabla} f_q$.

The third one holds by $\left\|\widetilde{\nabla} f\right\| \leq \xi$. For the fourth inequality, consider one term $\sum_{t=s}^{i-1}|\mathcal{S}_t| + |\mathcal{S}_{i,s}|$ for fixed i and $s \in \mathcal{S}_i$, which counts the number of feedbacks realized between iterations s and $i-1$, plus the ones realized in iteration i whose points were selected before iteration s. All these realized feedbacks were either generated before iteration s or between iterations s and i. Consider one of these feedbacks. Let p be the iteration in which this feedback was generated, and r be the iteration in which it is realized. We have either $s < p < i$ or $p < s$, but $r \in \{s, ..., i\}$. Similar to the argument made by [141], we then need to analyze two cases. For $s < p < i$, there are at most $i - s + 1 = D(s)$ possible indices for p, so we have $\sum_{s=1}^{T} D(s)$ feedbacks in total under this case. For $p < s$, it is easier to find the maximum possible indices for s instead of p. For each fixed p, there are at most $r - p + 1 = D(p)$ possible indices for s that are affected by p, so we have $\sum_{p=1}^{T} D(p)$ feedbacks in total under this case. Combining these two cases together, we have $\sum_{i=1}^{n}\sum_{s\in\mathcal{S}_i}\left(\sum_{t=s}^{i-1}|\mathcal{S}_t| + |\mathcal{S}_{i,s}|\right) \leq 2\sum_{i=1}^{n} D(i)$ in total.

Next, using the bounds developed in (2.24) and (2.25), we establish the following bound for Part I:

$$\begin{aligned}\sum_{i=1}^{n}\sum_{s\in\mathcal{S}_i}\langle \widetilde{\nabla} f_s, y_s - y^*\rangle &\leq \xi G\sqrt{n+D} + \xi\sum_{i=1}^{n}\sum_{s\in\mathcal{S}_i}\|y_s - y_{i,s}\| \\ &\leq \xi G\sqrt{n+D} + 2\,\xi^2\,\eta\,D \leq 3\,\xi G\sqrt{n+D}. \end{aligned} \tag{2.26}$$

Under assumption (2.4), we bound the total delay $D = \sum_{i=1}^{n} D(i)$ by applying a union bound:

$$\begin{aligned}\mathbb{P}\left(D \geq t\right) = \mathbb{P}\left(\sum_{i=1}^{n} D(i) \geq t\right) &\leq \mathbb{P}\left(\bigcup_{i=1}^{n}\left\{D(i) \geq \frac{t}{n}\right\}\right) \\ &\leq \sum_{i=1}^{n}\mathbb{P}\left(D(i) \geq \frac{t}{n}\right) \leq n\exp\left(-\frac{m^{p+1}}{\sigma_D^2}\right),\end{aligned}$$

where we choose $\frac{t}{n} = \mu_D + m$. If we then set the last expression of the above bound to δ and solve for m, the total delay is bounded by $D \leq n(\mu_D + \sqrt[p+1]{\sigma_D^2\log(n/\delta)})$ with probability $1-\delta$. Therefore, plugging this bound in (2.26), we derive the following for Part I, which holds with probability $1-\delta$:

$$\frac{1}{n}\sum_{i=1}^{n}\sum_{s\in\mathcal{S}_i}\langle \widetilde{\nabla} f_s, y_s - y^*\rangle \leq \left(\sqrt{1 + \mu_D + \sqrt[p+1]{\sigma_D^2\log(n/\delta)}}\right)\frac{3G\xi}{\sqrt{n}}. \tag{2.27}$$

Part II: To bound part II on the right-hand side of (2.23), we first define a filtration $\mathcal{F}_i$,

which is a sigma-field $\mathcal{F}_i = \sigma\left(\widetilde{\nabla} f_s : s \in \mathcal{S}_i\right)$ for each iteration i, and then have the following:

$$\begin{aligned}
&\mathbb{E}\Big[\langle\, \widehat{\nabla} f_i(y_i) - \widetilde{\nabla} f_i,\ y_i - y^*\rangle | \mathcal{F}_{i-1}\Big] = \mathbb{E}\Big[\langle\, \widehat{\nabla} f_i(y_i) - \widetilde{\nabla} f_i,\ y_i - y^*\rangle | y_i\Big] \\
&= (y_i - y^*)^T \cdot \mathbb{E}\Big[\big(\widehat{\nabla} f_i(y_i) - \widetilde{\nabla} f_i\big) | y_i\Big] = 0.
\end{aligned}$$

In the above, the first equality is because of the projected online stochastic sub-gradient descent step $y_i = \mathbf{Proj}_{\mathcal{K}}\big(y_{i-1} - \eta \sum_{s \in \mathcal{S}_{i-1}} \widetilde{\nabla} f_s\big)$, and the last equality is due to $\mathbb{E}\big[\widetilde{\nabla} f_i | \mathcal{F}_{i-1}\big] = \widehat{\nabla} f_i(y_i)$. So, the above result shows that the sequence $\big\{\langle\, \widehat{\nabla} f_i(y_i) - \widetilde{\nabla} f_i,\ y_i - y^*\rangle | \mathcal{F}_{i-1}\big\}_i$ is a *martingale difference sequence* (see Definition 4 in Appendix C).

Next, since $\widetilde{\nabla} f_i$ is a ρ-sub-Gaussian random variable (note that in Lemma II.3 in Appendix B, we show that this is the case for the dosage penalty function (2.2)), we have that $\langle\, \widehat{\nabla} f_i(y_i) - \widetilde{\nabla} f_i,\ y_i - y^*\rangle$ is a $G^2\rho$-sub-Gaussian random variable as well. By using Azuma-Hoeffding's inequality for this *sub-Gaussian martingale difference sequence* (see Theorem II.8 in Appendix C), we have:

$$\mathbb{P}\left(\frac{1}{n}\sum_{i=1}^{n} \langle\, \widehat{\nabla} f_i(y_i) - \widetilde{\nabla} f_i,\ y_i - y^*\rangle \ \ \geq\ t\right) \ \leq\ \exp\left(-\frac{nt^2}{2G^2\rho}\right),$$

which implies that

$$\frac{1}{n}\ \sum_{i=1}^{n} \langle\, \widehat{\nabla} f_i(y_i) - \widetilde{\nabla} f_i,\ y_i - y^*\rangle\ \leq \sqrt{\frac{2G^2\rho\log(1/\delta)}{n}}$$

holds with probability at least $1-\delta$. □

Remark. The S-SGD high-probability bound in Proposition 3 is developed under assumption of *delayed* feedbacks. If there were *no* delay in observing patient's feedbacks, then this loss bound would become $\sqrt{2G^2\rho\log(1/\delta)/n} + G\xi/\sqrt{n}$ with probability $1-\delta$. This high-probability loss bound helps us bound the S-SGD sub-optimality loss of $\mathbb{E}\big[\sum_{t=1}^{T}\big(\mathbf{V}_k^*(t) - \hat{\mathbf{V}}_k(t)\big)\big]$ in the Bayesian regret decomposition presented §2.4.3, which is related to learning the optimal dosage of any selected treatment k (see proof of Theorems II.1 and Corollary II.2).

2.4.6 Regret Analysis of the S-SGD Bandit Algorithm

In this section, we provide our theoretical result for deriving the *Bayesian regret* (Theorem II.1) of the proposed algorithm. Our proof for deriving the *regret* (Corollary II.2) is in Appendix B.

Proof. Proof of Theorem II.1: We show how each term of the Bayesian regret decomposition presented in §2.4.3 can be bounded using the results of Propositions 1, 2, and 3.

Part I (Contextual bandit loss): Proposition 2 developed a bound on the contextual bandit loss $\mathbb{E}\big[\sum_{t=1}^{T}\left(\mathbf{V}^*_{k^*}(t)-\mathbf{V}^*_k(t)\right)\big]$ (due to learning the optimal treatment k^*) with probability $1-2\delta$.

Part II (S-SGD sub-optimality loss): We first decompose this loss as follows:

$$
\begin{aligned}
&\mathbb{E}\Big[\sum_{t=1}^{T}\left(\mathbf{V}^*_k(t)-\hat{\mathbf{V}}_k(t)\right)\Big] \qquad (2.28)\\
&=\mathbb{E}\Big[\sum_{t=1}^{T}\Big(\sigma\left(\Phi_k^T(t)\cdot\theta-f_k\left(\Psi_k(t),y_k^*;\pi\right)\right)-\sigma\big(\Phi_k^T(t)\cdot\hat{\theta}(t)-f_k\left(\Psi_k(t),y_k^*;\hat{\pi}(t)\right)\big)\Big)\\
&+\sum_{t=1}^{T}\Big(\sigma\big(\Phi_k^T(t)\cdot\hat{\theta}(t)-f_k\left(\Psi_k(t),y_k^*;\hat{\pi}(t)\right)\big)-\sigma\big(\Phi_k^T(t)\cdot\hat{\theta}(t)-f_k\left(\Psi_k(t),y_k^*;\pi\right)\big)\Big)\\
&+\sum_{t=1}^{T}\Big(\sigma\big(\Phi_k^T(t)\cdot\hat{\theta}(t)-f_k\left(\Psi_k(t),y_k^*;\pi\right)\big)-\sigma\big(\Phi_k^T(t)\cdot\hat{\theta}(t)-f_k\left(\Psi_k(t),y_k(t);\hat{\pi}(t)\right)\big)\Big)\Big].
\end{aligned}
$$

The first term on RHS of (2.28) is bounded by Proposition 1 with probability $1-2\delta$ using a similar argument that we made in Part I of the proof of Corollary II.2 (see Appendix B) as:

$$
\begin{aligned}
&\mathbb{E}\Big[\sum_{t=1}^{T}\Big(\sigma\left(\Phi_k^T(t)\cdot\theta-f_k\left(\Psi_k(t),y_k^*;\pi\right)\right)-\sigma\big(\Phi_k^T(t)\cdot\hat{\theta}(t)-f_k\left(\Psi_k(t),y_k^*;\hat{\pi}(t)\right)\big)\Big)\Big]\\
&\leq\frac{L(T,\delta)+Q(T,\delta)}{2},
\end{aligned}
$$

where $L(T,\delta)$ and $Q(T,\delta)$ are defined in Proposition 2.

Using the Lipschitz property of the logistic function with constant $1/4$ and the penalty function with constant $\max\{\alpha_k,\beta_k\}$, the second term on RHS of (2.28) is bounded with probability $1-\delta$ as:

$$
\begin{aligned}
&\mathbb{E}\Big[\sum_{t=1}^{T}\Big(\sigma\big(\Phi_k^T(t)\cdot\hat{\theta}(t)-f_k\left(\Psi_k(t),y_k^*;\hat{\pi}(t)\right)\big)-\sigma\big(\Phi_k^T(t)\cdot\hat{\theta}(t)-f_k\left(\Psi_k(t),y_k^*;\pi\right)\big)\Big)\Big]\\
&\leq\frac{\max\{\alpha_k,\beta_k\}}{4}\,\mathbb{E}\Big[\sum_{t=1}^{T}\Big(\Upsilon_k^T(t)\cdot\pi-\Upsilon_k^T(t)\cdot\hat{\pi}(t)\Big)\Big]\leq\frac{Q(T,\delta)}{2},
\end{aligned}
$$

where the bound for (2.12) that we developed in the proof of Proposition 1 is used in the above.

The third term on RHS of (2.28) is bounded by Proposition 3 with probability $1-2\delta$ as:

$$\mathbb{E}\Big[\sum_{t=1}^{T}\Big(\sigma\big(\Phi_k^T(t)\cdot\hat{\theta}(t)-f_k\left(\Psi_k(t),y_k^*;\pi\right)\big)-\sigma\big(\Phi_k^T(t)\cdot\hat{\theta}(t)-f_k\left(\Psi_k(t),y_k(t);\hat{\pi}(t)\right)\big)\Big)\Big]$$
$$\leq\frac{1}{4}\mathbb{E}\Big[\sum_{t=1}^{T}\Big(f_k\left(\Psi_k(t),y_k^*;\pi\right)-f_k\left(\Psi_k(t),y_k(t);\hat{\pi}(t)\right)\Big)\Big]\leq P(\delta,T)\sqrt{T},$$

where $P(\delta,T)=\frac{1}{4}G\sqrt{2\rho\log(1/\delta)}+\frac{3}{4}\xi G\sqrt{1+\mu_D+\sqrt[p+1]{\sigma_D^2\log(T/\delta)}}$. Note that in above we use Lemma II.3 (see Appendix B) in which we prove that the dosage penalty function (2.2) has ρ-sub-Gaussian sub-gradients and the second moment of these sub-gradients is bounded by ξ. Also, G is the diameter of the dosage decision space, and we have

$$y_k^*=\underset{y_k\in[\Delta_k^{LB},\Delta_k^{UB}]}{\arg\min}\sum_{t=1}^{T}f_k\left(\Psi_k(t),y_k;\pi\right)$$

.

Inserting the above three high-probability bounds in (2.28), the following bound is obtained with probability $1-5\delta$ to bound the S-SGD sub-optimality loss:

$$\mathbb{E}\Big[\sum_{t=1}^{T}\big(\mathbf{V}_k^*(t)-\hat{\mathbf{V}}_k(t)\big)\Big]\leq P(\delta,T)\sqrt{T}+\frac{L(T,\delta)}{2}+Q(T,\delta).$$

Part III (Estimation loss): We bound the total estimation loss $\mathbb{E}\big[\sum_{t=1}^{T}\big(\hat{\mathbf{V}}_k(t)-\mathbf{V}_k(t)\big)\big]$ with probability at least $1-2\delta$ using Proposition 1 and applying a similar argument that we developed in Part I of the proof of Corollary II.2 (see Appendix B) as follows:

$$\mathbb{E}\Big[\sum_{t=1}^{T}\big(\hat{\mathbf{V}}_k(t)-\mathbf{V}_k(t)\big)\Big]\leq\frac{L(T,\delta)+Q(T,\delta)}{2}.$$

Putting all the high-probability bounds that we established in the above three parts together derives the following Bayesian regret:

$$\textbf{BayesianRegret}(T)\leq P(\delta,T)\sqrt{T}+2L(T,\delta)+\frac{5Q(T,\delta)}{2}+2T\delta,$$

which completes the proof when we set $\delta=1/T$ in the above bound. □

2.4.7 Extension: Contextual Learning with Bandit Convex Optimization

In §2.4.5, we analyzed the proposed algorithm and developed online S-SGD to adaptively optimize the dosage of each treatment (Proposition 3). We now generalize our theoretical result to the case, where the decision-maker does not necessarily have access to a gradient oracle at any point in the space. To this aim, we first develop a high-probability regret bound for a sub-Gaussian *Bandit Stochastic Gradient Descent* (B-SGD) procedure (see Proposition 5 in Appendix B). We then derive the Bayesian regret of the B-SGD Bandit algorithm in which we use B-SGD instead of S-SGD (see Algorithm 3 in Appendix E), below (see the proof in Appendix B).

Proposition 4 (**Bayesian Regret of the TS-based B-SGD Bandit Algorithm**). The Bayesian regret of the TS-based B-SGD Bandit algorithm is $\widetilde{\mathcal{O}}\left(T^{3/4}(1+\mu_D)^{3/4}\right)$ under delayed feedback, and $\widetilde{\mathcal{O}}\left(T^{3/4}\right)$ under immediate feedback (no delay) over finite horizon T.

2.5 Case Study and Empirical Results

Our model and methodology are motivated by a pressing clinical question in regard with managing risk of Cardiovascular Disease (CVD) in patients with Type 2 Diabetes Mellitus (T2DM), namely the choice and appropriate dosage of Blood Pressure (BP)-lowering agents. Using recent clinical trial data, we evaluate the performance of the S-SGD Bandit algorithm compared with the physicians' decisions made in the clinical trial and four benchmark policies.

2.5.1 BP Control for T2DM Patients at High Risk for Cardiovascular Events

Background. T2DM is a chronic condition characterized by high sugar levels in blood and affects the way the patient's body metabolizes sugar or glucose (an important source of fuel for the body). With T2DM, the patient's body either resists the effects of insulin (a hormone that regulates the movement of sugar into cells), or does not produce enough insulin to maintain normal glucose levels [23]. High Blood Pressure (HBP), or *hypertension* is a common condition in T2DM patients, where the blood is pumped through the heart and blood vessels with excessive force. Over time, HBP tires the heart muscle and can enlarge it. It substantially increases the risk of both macrovascular and microvascular complications (e.g., stroke, retinopathy, neuropathy, peripheral vascular and coronary artery diseases) [14]. Treating and managing HBP is critical in preventing these and other diabetes complications. According to new statistics by the American Heart Association, about 103 million U.S. adults have HBP, contributing cause of death for more than 1,100 daily deaths and costing the

nation $48.6 billion yearly [24].

BP is assessed using two measures. The top number is the *systolic blood pressure* (SBP), which refers to the pressure inside the artery when the heart contracts and is pumping the blood through the body. The bottom number is the *diastolic blood pressure* (DBP), which refers to the pressure inside the artery when the heart is at rest and is filling with blood. The most recent guidelines by the American Heart Association/American College of Cardiology (AHA/ACC) identify an SBP/DBP of 120/80 mmHg as "normal" BP. They recommend a target BP of 130/80 mmHg for patients with hypertension, and patients at high risk for CVD may benefit from a more stringent target of 120/80 mmHg [160]. The American Diabetes Association similarly recommends a tight BP target (e.g., 120/80 mmHg) for patients with diabetes at high CVD risk [3].

Research Motivation and Question. There are a number of medication classes for lowering BP; each class includes several different medications at various dosages. Treatment of HBP usually starts with a single BP-lowering agent, and subsequent agents can be added if the target is not met. In patients with T2DM at high risk for CVD, *multiple-drug* therapy, often with three or four BP-lowering agents, is needed to achieve the BP target ([14] and [3]).

There is a consensus about the choice of the first two lines of BP-lowering medications based on their effectiveness in preserving renal function and reducing CVD risk. The *first-line* medication class is Angiotensin-Converting Enzyme (ACE) inhibitors, or Angiotensin II Receptor Blockers (ARBs) if the patient is unable to take an ACE. The *second-line* medication class is Dihydropyridine (DHP) Calcium-Channel Blockers (CCB) or thiazide-type diuretics ([3], [43], and [134]). Clinical guidelines, however, *lack* clarity on the third-line medication, which many patients with T2DM will eventually need to achieve a controlled BP.

In the case study, we thus focus on the *third-line* BP-lowering medication. Inspired by this clinical question, our *main research question* is how joint online learning and optimization algorithms can help medical professionals find the third-line medication and its dosage for effective BP control in patients with T2DM at high risk of CVD. We consider SBP of 120 mmHg as the target based on the most recent clinical guidelines on BP control for these patients [3].

Remark. Our methodology is more general than the version treated in the case study. To illustrate a richer medical problem that can be addressed by our model, consider the T2DM patients who are at elevated risk of Chronic Kidney Diseases (CKD). Having HBP may damage the kidneys and deteriorate kidney function. CKD may eventually lead to kidney failure requiring dialysis or a kidney transplant to maintain life. Glomerular filtration rate

(GFR), estimated from age, sex, race, and a blood test [89], is used as a measure of patient's kidney function. If GFR is too low (often < 60 mL/min/1.73 m^2), the patient has moderate CKD. This implies that kidneys are not able to remove enough waste and extra fluid from blood; thus, clinicians prescribe a BP-lowering medication (either an ACE-inhibitor, or an ARB) that is known to have reno-protective effects ([13]). For this medical problem, the aim is to keep GFR high (> 60 mL/min/1.73 m^2) by recommending a BP-lowering medication (all medications in the ACE-inhibitor and ARB classes) and its dosage. Accordingly, the main outcome can be defined as whether GFR is greater than 60 or not, and the treatment-dosage sub-outcome can be SBP. Our model could help doctors adaptively learn a treatment regimen given the contextual patient information to control GFR through prescribing a BP-lowering medication and its dosage. Since we lack data on CKD, our case study instead focuses on choosing medications for patients with T2DM at high risk of CVD.

2.5.2 Data Description and Problem Formulation

The Action to Control Cardiovascular Risk in Diabetes (ACCORD) Blood Pressure (ACCORD BP) clinical trial studied whether lowering SBP to 120 mmHg (intensive therapy) versus 140 mmHg (standard therapy) reduces major cardiovascular events. These were defined as nonfatal myocardial infarction, nonfatal stroke, or death from cardiovascular causes captured in 4,700 patients with T2DM at high CVD risk. Patients treated with intensive SBP therapy experienced fewer major cardiovascular events (hazard ratio, 0.88), though it did not reach statistical significance after 4.7 years mean follow-up time (95% confidence interval for hazard ratio, 0.73-1.06) [78].

ACCORD enrolled 10,251 patients with T2DM at high risk for CVD. Of those, 4,733 participants were randomly assigned to either intensive or standard BP control (ACCORD BP) to study the effects of intensive versus standard therapy on diabetes related outcomes. We focus on 2,012 patients of these with a total of 29,447 visits in ACCORD BP that were assigned to intensive therapy. For such patients, BP was measured every month for 4 months and once every 2 months thereafter, but additional visits occurred as needed to monitor and assure appropriate implementation of the study intervention. The intensive BP therapy targets a SBP of 120 mmHg. Whenever SBP was not on the target, the trial called for either the medication dose to be titrated to a very precise (continuous) dosage, or the addition of one or more medications.

In consultation with medical researchers and clinicians at Harvard Medical School, we included 62 covariates that are believed to be relevant to the task of finding the right BP medication and dosage. Those include the following:

- *Demographics*: Age, gender, and race group.
- *Pre-existing conditions*: History of CVD, and cigarette smoking in the last 30 days.
- *Measurements*: SBP, DBP, heart rate (HR), glycosylated hemoglobin (HbA1c), total cholesterol, triglycerides, very low/low/high density lipoprotein (VLDL, LDL, and HDL), fasting plasma glucose (FPG), alanine aminotransferase (ALT), potassium, serum creatinine (SCr), estimated glomerular filtration rate (eGFR), weight, and height.
- *Medications and dosages*[2]: ACE-inhibitor (benazepril, lisinopril, and ramipril), ARB (candesartan and valsartan), CCB-DHP (felodipine and amlodipine), CCB non-DHP (diltiazem), Beta-Blocker (metoprolol), Diuretic (chlorthalidone and HCTZ), Loop Diuretic (furosemide), Alpha-Beta Blocker (carvedilol), Alpha Blocker (terazosin), Vasodilator (hydralazine), and Sympatholytic (reserpine). The dosage of each prescribed medication is also available.
- *Adherence to medications*: Percentage of the time that the patient adhered to each prescribed medication with a specified dosage since their last visit.

As described in §2.5.1, to achieve the SBP target of 120 mmHg, doctors initially prescribed a *first-line medication* from the ACE-inhibitor, or ARB classes. Next, they often prescribed a *second-line medication* from the CCB-DHP, or CCB non-DHP classes. Nonetheless, many patients require additional BP-lowering medications to achieve the target BP, and clinical guidelines are not specific on the choice of *third-line medication*. Table 2.1 shows the list of all third-line medication classes in the clinical trial and their medications with their possible dosage ranges. A doctor must select one of these medications with a specified dosage in order to achieve the SBP target for an individual.

Contextual bandit with a two-dimensional control model. We formulate the problem of finding an appropriate third-line BP-lowering medication and its dosage as a contextual MAB with a two-dimensional control problem, which is defined by the following elements:

(a) *Arms* (*first decision*): As listed in Table 2.1, we have 8 possible medications that can be treated as our first decision variable (an 8-armed contextual bandit problem).

(b) *Dosage* (*second decision*): Each of the 8 medications in Table 2.1 has a pre-specified range of possible dosages, which is continuous because they titrated these dosages finely.

[2]Note that the names inside the parentheses are the medication names and the names outside the parentheses are the medication class. We may have more than one medication from each medication class, as well. We also have the particular dosage information of each prescribed medication.

Medication class	Medication name	Dosage range
Beta-Blocker	Metoprolol	[25, 200]
Diuretic	HCTZ	[12.5, 25]
Diuretic	Chlorthalidone	[7.5, 25]
Loop Diuretic	Furosemide	[10, 80]
Alpha-Beta Blocker	Carvedilol	[3.125, 25]
Alpha blocker	Terazosin	[1, 10]
Sympatholytic	Reserpine	[0.125, 0.345]
Vasodilator	Hydralazine	[25, 100]

Table 2.1: The list of all medication classes with their medication names in each class and corresponding possible dosage ranges that can be used as the third-line medication to control the SBP target.

(c) *Contextual information*: There are 62 patient-specific covariates described above.

(d) *Treatment-dosage sub-outcome*: This metric is SBP. Doctors measure SBP based on the average of three measurements using an automated device. The threshold parameter q is 120.

(e) *Expected reward*: This includes only the dosage penalty function that needs to be minimized. It can be measured by how much SBP is greater or less than the threshold of 120.

It is worth noting that evaluating online algorithms retrospectively on the observational data is a challenging task, because it requires access to counterfactuals in some scenarios. For instance, assume that our online algorithm chooses "Chlorthalidone" as a third-line medication for a patient, while the third-line medication prescribed for the patient in the data set is "Terazosin." In such a scenario, we need to know the feedback associated with prescribing "Chlorthalidone" to the patient to be able to evaluate our online algorithm's performance. To deal with this issue, we estimate the counterfactuals associated with the treatment-dosage sub-outcome using all the observational data. Specifically, we fit a linear regression model to predict the treatment-dosage sub-outcome corresponding to each of the 8 medications, and use this estimation in our performance evaluation analysis. We also require the optimal dosage of each medication for our algorithm's evaluation. We find the optimal dosage of each medication by minimizing the sum of penalty functions over the patients who were prescribed this medication. Note that the unknown coefficients of penalty functions are estimated using a linear regression over all the data.

2.5.3 Evaluation and Empirical Results

Using the ACCORD BP, we evaluate the performance of our algorithms compared to four benchmark policies and the physicians' decisions made in the trial. We use two performance measures: cumulative regret and distribution of success rate. The *cumulative regret* measures the difference between the (expected) cumulative rewards of the clairvoyant optimal policy and our algorithm. The *distribution of success rates* describes the distribution of the total success rate by the end of the time horizon. The *success* event is defined as having a SBP level of between 115 and 125 mmHg. We conduct an additional analysis to assess the impact of delayed feedback on our performance measures. Lastly, we compare our algorithms with benchmarks in terms of medication decisions.

Since we have a two-dimensional control feature, we cannot directly compare our algorithm with most bandit algorithms in the literature. We consider 10 random permutations of our data set and compare the following algorithms together:

(a) TS-SGD algorithm: This is the TS-based S-SGD Bandit algorithm described in §2.3.2.

(b) UCB-SGD algorithm: This is the UCB-based S-SGD Bandit algorithm (see Appendix D).

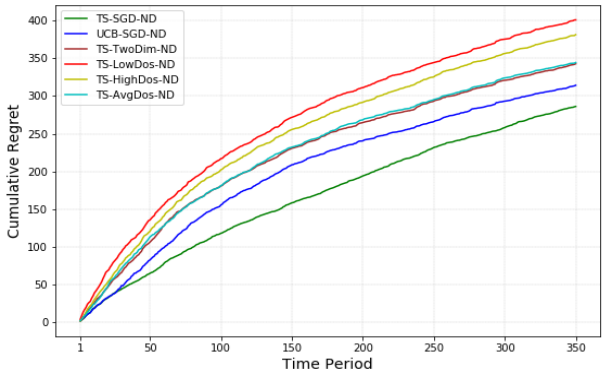
(c) Fixed-dosage algorithms: They do not involve dosage optimization; rather each uses a fixed (low/average/high) dosage for a selected medication, and the medication decision is made by a contextual TS algorithm. We denote them by TS-LowDos, TS-AvgDoc, and TS-HighDos.

(d) TS-TwoDim algorithm: This policy considers an arm for each pair of treatment (one of the 8 medications in Table 2.1) and dosage (grouped as being either low, average, or high). This is indeed a 24-armed bandit, and a contextual TS is used to make medication-dosage decisions.

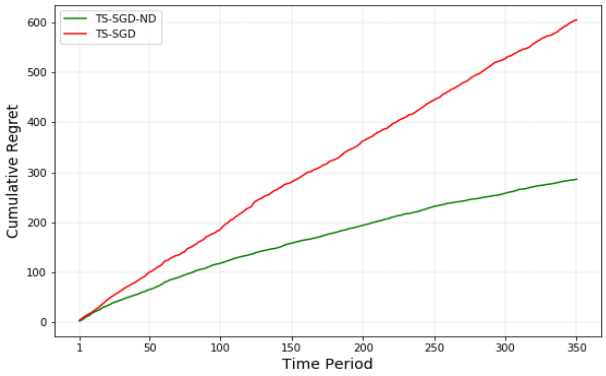
Note that the reason why we have chosen the low, average, and high dosages in our benchmarks is that these dosage selection policies are very common in medical practice. In particular, the high complexity of selecting an optimal dosage based on unique patient characteristics is a barrier; thus, doctors usually employ these dosage for a selected treatment, rather than optimizing the dosage.

Cumulative Regret. Figure 2.3 illustrates the cumulative regret of each algorithm as a function of time period. To exclude the impact of delayed feedback on our comparison, we first assumed that patients' feedback is realized immediately after prescribing a treatment with a dosage. For each of the above algorithms, Alg-ND refers to an algorithm under *no*

delayed feedback. As seen in Figure 2.3a, our TS-SGD-ND is the fastest learning one followed by UCB-SGD-ND across all time periods. TS-TwoDim-ND and TS-AvgDos-ND have the next best performance. Among the fixed-dosage policies, the average, high, and low ones are ordered in decreasing performance. Figure 2.3a illustrates the importance and effect of optimizing the dosage decision on the cumulative regret. Even though all these policies deploy either TS or UCB for choosing medication, only TS-SGD-ND and UCB-SGD-ND have an optimization procedure for finding the best dosage. Hence, having a systematic online optimization procedure for optimizing the dosage decision can help clinicians make more informed decisions for treating chronic diseases.



(a) Different online algorithms under no delay.



(b) TS-SGD algorithm under delay and no delay.

Figure 2.3: Performance evaluation of different online learning algorithms in terms of cumulative regret.

Effect of Delayed Feedback. In practice, the delay in observing patient feedback in ACCORD BP ranges from two weeks to one month. We introduced an *on the fly* strategy in our algorithms to deal with this delayed feedback. It exploits the information of the patients for whom a treatment with a specified dosage are already prescribed and their feedback is revealed up to the current time. Figure 2.3b compares the cumulative regret of our TS-SGD under no delay feedback and delayed feedback. As expected, TS-SGD-ND outperforms TS-SGD, because it leverages the information to which we do not have access in practice.

Distribution of Success Rate. Figure 2.4 illustrates the distribution of the success rate for different algorithms over 500 time periods with 10 random permutations of data. We evaluate the success at *every patient visit* by checking whether the predicted SBP is between 115 and 125 mmHg for the selected medication and dosage. We observe that TS-SGD and UCB-SGD achieve a median success rate of 80% and 77%, respectively, whereas the four benchmark policies have significantly a lower median success rate of around 71%. For the ACCORD BP trial, all permutations yield a 72% median success rate. This observation

highlights the benefits of making dosage decisions adaptively and optimally through our online optimization procedure.

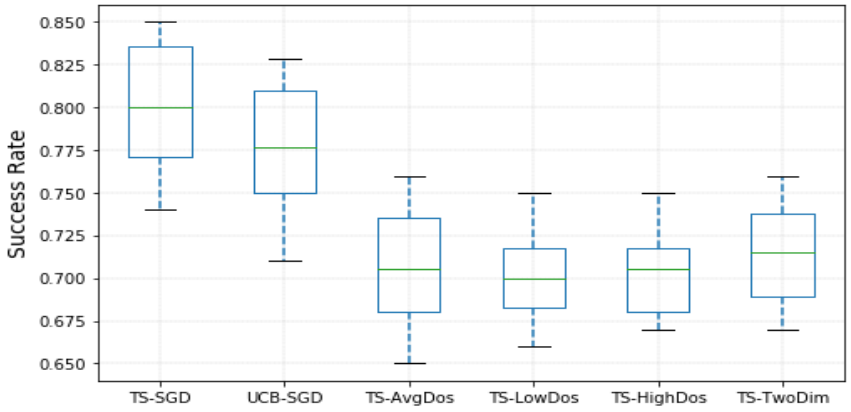


Figure 2.4: Distribution of success rate for different online learning algorithms over 500 time periods.

Next, we evaluate and compare the algorithms in terms of percentages of both success and failure decisions in choosing the best medication and its dosage. Here, we look at the *last visit* of every patient and check whether the predicted SBP is between 115 and 125 mmHg at the last visit. Table 2.2 presents the percentages of both success and failure rates for the ACCORD BP trial, our proposed algorithms (TS-SGD and UCB-SGD), TS-TwoDim, and three fixed-dosage algorithms (TS-AvgDos, TS-LowDos, and TS-HighDos). We find that our algorithms (TS-SGD and UCB-SGD) have higher success and lower failure rates in making medication and dosage decisions compared to what the doctors did in the trial. For example, TS-SGD makes a successful medication and dosage decision 80.25% of the times (i.e., in 80.25% of the visits, SBP will be between 115 and 125 mmHg), in contrast to the 71.26% success rate achieved in the trial.

	Trial	**TS-SGD**	**UCB-SGD**	**TS-TwoDim**	**TS-AvgDos**	**TS-LowDos**	**TS-HighDos**
Success	71.2%	80.2%	78.7%	73.5%	72%	71.5%	71.7%
Failure	28.7%	19.7%	21.2%	26.4%	27.9%	28.4%	28.2%

Table 2.2: The percentages of both success and failure in the BP ACCORD trial and online learning algorithms in choosing successful medications and dosages (i.e., success is defined as having $\mathbf{115} \leq \text{SBP} \leq \mathbf{125}$ mmHg).

Medication Decisions. We next analyze how our algorithms make decisions compared to the medications offered in the ACCORD BP trial and benchmark policies. To this aim, we divide the data set into two subsets, including (i) success data subset: the data points at which a medication and its dosage were prescribed in the trial such that it resulted in having

$115 \leq \text{SBP} \leq 125$ mmHg; and (ii) failure data subset: the data points at which a medication and its dosage were prescribed such that it resulted in having a SBP out of the range of 115-125 mmHg. We run our algorithms on the whole data set and analyze the results for both success and failure data subsets separately.

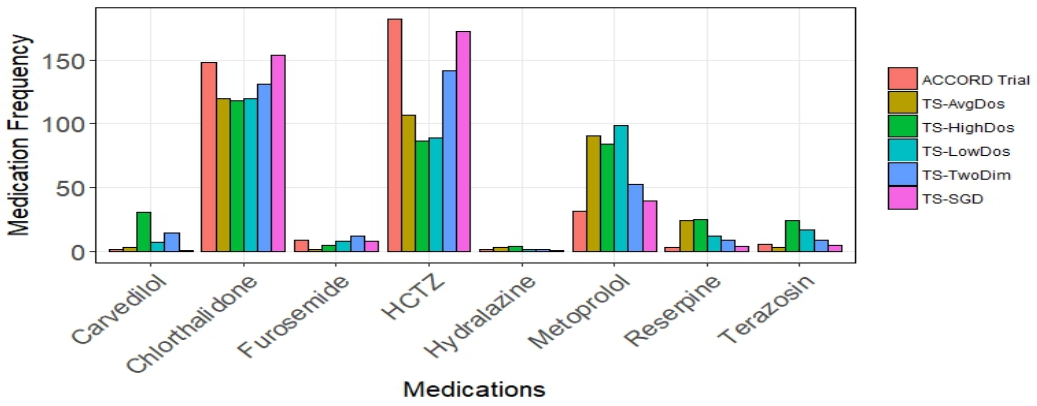


Figure 2.5: The frequency of selected medications over the success data subset with the policies ordered left to right (ACCORD Trial, TS-AvgDos, TS-HighDos, TS-LowDos, TS-TwoDim, and TS-SGD).

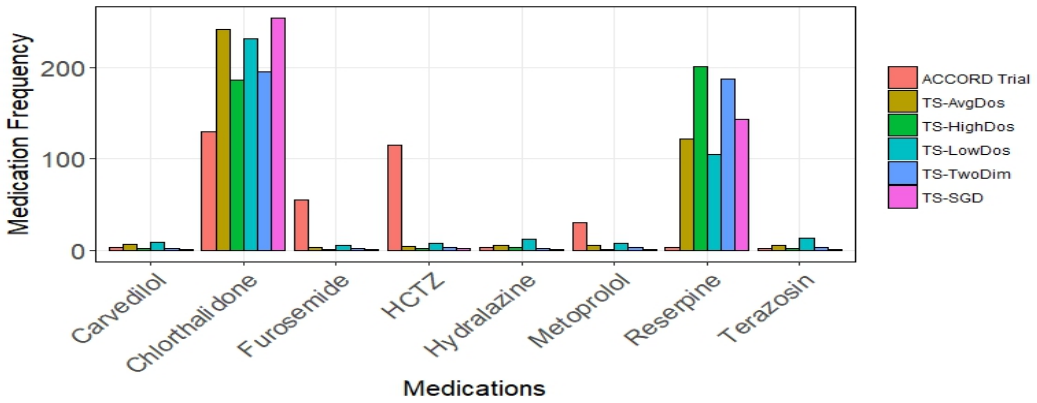


Figure 2.6: The frequency of selected medications over the failure data subset with the policies ordered left to right (ACCORD Trial, TS-AvgDos, TS-HighDos, TS-LowDos, TS-TwoDim, and TS-SGD).

Figures 2.5 and 2.6 illustrate the frequency of medications chosen in the ACCORD BP trial, the four benchmarks, and TS-SGD for the visits in success and failure data subsets, respectively (see Tables 2.5 and 2.6 in Appendix G for the percentages). The key observation from this analysis is that for the *success* data subset, our algorithm (TS-SGD) behaves *similarly* to the trial in choosing medications. For example, both the trial and TS-SGD choose "Chlorthalidone" and "HCTZ" more often than other medications in the success

data subset. In particular, "Chlorthalidone" was selected 39.7% (38.5%) of the times, and "HCTZ" was selected 44.4% (47.3%) of the times by TS-SGD (the trial). However, in the *failure* data subset, TS-SGD chooses medications *differently* from what was done in the trial for half the medications. For example, TS-SGD chooses "Chlorthalidone" and "Reserpine" more often than other medications in the failure data subset, which is quite different than the medications selected in the trial. In particular, "Chlorthalidone" was selected 55.2% (37.8%) of the times, "Reserpine" was selected 31.4% (1.2%) of the times, "HCTZ" was selected 6.9% (33.4%) of the times, and "Furosemide" was selected 1.8% (16%) of the times under TS-SGD (the trial).

We next investigate how TS-SGD may achieve *success* (i.e., a predicted SBP of 115-125 mmHg) by suggesting a different BP-lowering medication in patient visits, where the clinical trial has *failed* (i.e., patient had a SBP out of 115-125 mmHg). To this aim, we focus on a subset of patient visits in which the clinical trial failed and TS-SGD suggests a different medication. We then assess whether the different medication selected by TS-SGD results in a success. TS-SGD deviates from the trial's medication selections in 38.1% of the cases in the failure data subset. Within this group, Table 2.3 shows the distribution of medication choices TS-SGD made to convert the failure decisions in the trial into predicted successes by selecting alternative medications.

Reserpine	**Chlorthalidone**	**Metoprolol**	**Furosemide**	**Terazosin**	**Carvedilol**	**Hydralazine**	**HCTZ**
28.2%	43%	2.5%	8.2%	3.5%	2.4%	1.8%	10.4%

Table 2.3: The distribution of different medications selected by TS-SGD conditioned on the successful outcomes when tested on the subset of failure data in which clinicians erroneously predicted success.

We obtain the insight that, as a result of modifying the decisions in the cases, where BP control was not achieved in the trial that occurred 28.7% of the times (see Table 2.2), 38.1% of these failures were *avoided* according to our TS-SGD algorithm. Note that TS-SGD has also a success rate of 97.2% over the success data subset. This gives the insight that TS-SGD did very well in the cases for which the trial outcome was achieved. Putting these together, the total success rate of TS-SGD reaches 80.2%. This reflects a predicted relative improvement of 12.6% with respect to the success rate of 71.2% in the trial.

2.5.4 Clinical Insights and Discussion

We discuss four important clinical insights from our analysis below.

First, in the current practice of treating and managing HBP, physicians usually deploy a *trial-and-error* approach to select a third-line medication and its dosage. These decisions are highly informed by the physician's expertise and experience of administrating various

medications and dosages, because the current guidelines do not inform the choice of the third-line medication, which many T2DM patients require to achieve the target BP. Instead, our (joint) contextual learning and optimization framework provides a systematic decision support tool for optimizing the medication and dosage decisions for these patients. Our empirical results shown in Figures 2.3 and 2.4 and Table 2.2 suggest that choosing dosage decisions adaptively and optimally following an online optimization procedure may improve clinical decisions for the HBP therapy. Of course, with offline data, offline methods can perform at least as well; however, we are using the data in an online manner to give insight into the potential performance of online learning and control.

Second, our results in Figures 2.5 and 2.6 as well as Tables 2.5 and 2.6 suggest that further attention should be given to "Chlorthalidone", "Reserpine", "Furosemide", and "HCTZ" (in decreasing order of discrepancy from the choice made in the trial) to (i) better understand the precise characteristics of patients that lead to the failures/successes in achieving the SBP target and (ii) confirm if the choices of the TS-SGD algorithm are indeed better. For the success data subset, our algorithm matches the trial in choosing "Chlorthalidone" and "HCTZ" more often than other medications, which we take as strong evidence that the TS-SGD algorithm has face validity. However, in the failure data subset of Table 2.6, "Furosemide", and "HCTZ" are used less frequently by our algorithm than in the trial; thus, it is critical to investigate why physicians used them more frequently in the trial than our algorithms. Further research should examine if our algorithm's confidence in prescribing "Chlorthalidone" and "Reserpine" more frequently bears out in clinical practice. We acknowledge that there is uncertainty around the true outcomes with model-suggested medications, because we lack data on treatments not selected in the trial (for obvious reasons).

Third, we show in Table 2.3 that for the failure data subset, TS-SGD improves HBP management by alternative medication selections at the proper dosage and converts the failures into successes, defined as $115 \leq$ predicted SBP ≤ 125 mmHg, in 38.1% of the cases. We also find that for these successful cases, "Chlorthalidone" and "Reserpine" are selected for 43% and 28.2% of the times, respectively. These are not far off from the average percentage of the times they were selected by TS-SGD over the entire failure data subset, which gave 55.2% and 31.4%, respectively, in Table 2.6.

Fourth, our empirical results provide medical professionals with critical insights into the effect of different third-line medications on achieving the BP target of 120 mmHg for patients with type 2 diabetes. To find the right medication and its corresponding dosage to achieve BP control, instead of a trial-and-error approach that is common in everyday clinical practice, our decision support tool provides an optimized medication and dosage considering all key contextual variables of a given individual. Personalizing BP treatment, in particular for

patients with diabetes who are at increased risk of cardiovascular and microvascular events, can improve outcomes and result in cost containment by averting some of these costly events.

2.6 Conclusion

We introduced a new contextual multi-armed bandit model with a two-dimensional nested control, and proposed the first joint contextual learning and optimization algorithm for it. For this algorithm, we provided a rigorous analytical performance analysis that involves several new technical ideas integrating the strength of contextual bandit and online convex optimization in a seamless fashion. Although our model and algorithm were motivated by a fundamental medical decision-making problem, they can also be applied to a wide range of other operation problems (e.g., joint inventory and pricing/vehicle routing) in which there are two levels of decision-making process.

Our algorithm/model fills a fundamental and important gap in the medical decision-making literature. We illustrated our algorithm's practical relevance by evaluating it on a critical chronic disease problem of controlling high blood pressure for patients with type 2 diabetes. We addressed a key need in current clinical guidelines of blood pressure management, namely that they do not inform the choice and dosage for the third-line BP-lowering medication to maintain the target SBP of 120 mmHg. In §2.5.3, we found that despite the losses due to online learning, our algorithm achieved 12% higher success rate than the clinical trial achieved. In §2.5.4, we provided four important insights in treating and managing HBP. In visits for which the trial did not achieve the SBP target, the alternative medications that were successfully selected most often by our algorithm were "Chlorthalidone" and "Reserpine" in 43% and 28.2% of cases, respectively. Similarly, "HCTZ" and "Furosemide" exhibited this pattern but were used less frequently. These medications deserve further study, because lacking access to counterfactual outcomes for these cases prevents us from forming a statistically strong conclusion, as is often the case in online learning, particularly in healthcare applications. Our findings also suggest that in almost 29% of cases, the medication and dosage choices made in the clinical trial were not effective. While our algorithms largely agreed with the medications used in the clinical trial for cases with successful BP control, we were able to identify promising alternatives for cases that did not achieve the BP control in the clinical trial.

2.7 Appendix

2.7.1 Appendix A: Summary of Major Notation

Table 2.4 summarizes the major mathematical notation used in the manuscript.

Notation	Description
$t \in \mathcal{T}$	index of time periods (correspond to epochs of patient arrivals).
$k \in \mathcal{K}$	index of actions (or treatments/medications).
$\phi^{\mathcal{X}}(t)$	patient t's context that directly affects disease progression risk.
$\psi^{\mathcal{X}}(t)$	patient t's context that affects disease progression risk via treatment-outcome metric.
$\phi_k^{\mathcal{A}}(t)$	action vector of patient t.
$y_k(t)$	dosage vector corresponding to treatment $k \in \mathcal{K}$ for patient t.
Δ_k^{LB}	the lower bound for a dosage of treatment $k \in \mathcal{K}$.
Δ_k^{UB}	the upper bound for a dosage of treatment $k \in \mathcal{K}$.
$\Phi_k(t)$	feature vector of patient t that directly affects the disease progression risk.
$\Psi_k(t)$	feature vector of patient t that affects the disease risk via treatment-outcome metric.
$\Upsilon_k(t)$	feature-dosage vector of patient t including the feature $\Psi_k(t)$ and dosage $y_k(t)$ vectors.
$\Pi_k(t)$	treatment–outcome metric of patient t.
$\zeta_k(t)$	the error term of the treatment–outcome metric of patient t for treatment k.
$R_k(t)$	reward (probability of not having disease progression) of patient t for treatment k.
$\xi_k(t)$	the error term of the reward of patient t for treatment k.
q	the target for the treatment-outcome metric.
α_k, β_k	weighs for the over-treatment and under-treatment, respectively.
π	true parameter in the online linear regression for treatment–outcome metric.
$\tilde{\pi}(t)$	sampled π at time period t from the posterior distribution.
$\bar{\pi}(t)$	estimated π at time period t from linear regression for treatment–outcome metric.
$\hat{\pi}(t)$	projected $\bar{\pi}(t)$ onto the admissible set Λ at time period t.
θ	true vector parameter in logistic regression for disease progression risk.
$\tilde{\theta}(t)$	sampled θ at time period t from the posterior distribution.
$\bar{\theta}(t)$	estimated θ at time period t from logistic regression for disease progression risk.
$\hat{\theta}(t)$	projected $\bar{\theta}(t)$ onto the admissible set Θ at time period t.
$\mathcal{H}_t$	history of observed information at the beginning of time period t.
m_i^t	mean of the Gaussian posterior distribution for the i-th element of θ at time period t.
$(q_i^t)^{-1}$	variance of the Gaussian posterior distribution for the i-th element of θ at time period t.
u^t	mean of the Gaussian posterior distribution of π at time period t.
$(P^t)^{-1}$	variance of the Gaussian posterior distribution of π at time period t.
$\widetilde{\nabla} f_k(t)$	stochastic (noisy) sub-gradient of function f at time period t for treatment k.
$D(t)$	The stochastic delay in observing the feedback of patient t.
$UB_k(t)$	Upper bound on the disease progression risk for treatment k at time period t.
$LB_k(t)$	Lower bound on the disease progression risk for treatment k at time period t.

Table 2.4: Summary of major notation in the manuscript.

2.7.2 Appendix B: Omitted Theoretical Results and their Proof

Proof. Proof of Corollary II.2: Similar to the decomposition that we introduced in §2.4.3 for the Bayesian regret, we can have the following decomposition for the regret:

$$\textbf{Regret}(T,\theta,\pi)=\sum_{t=1}^{T}\Big(\mathbf{V}^{*}_{k^*}(t)-\mathbf{V}^{*}_{\tilde{k}}(t)\Big)+\sum_{t=1}^{T}\Big(\mathbf{V}^{*}_{\tilde{k}}(t)-\hat{\mathbf{V}}_{\tilde{k}}(t)\Big)+\sum_{t=1}^{T}\Big(\hat{\mathbf{V}}_{\tilde{k}}(t)-\mathbf{V}_{\tilde{k}}(t)\Big),$$

where $\tilde{k}$ is the treatment chosen by the UCB-based S-SGD Bandit algorithm (see Algorithm 2 in Appendix D) at time period t and k^* is the optimal treatment. We next show how each term of the above regret decomposition can be bounded below.

Part I (Contextual bandit loss): Consider the term $(\mathbf{V}^{*}_{k^*}(t)-\mathbf{V}^{*}_{\tilde{k}}(t))$ as the loss corresponding to the difference between the true expected rewards of the optimal treatment k^* and the treatment $\tilde{k}$ chosen by the algorithm at time period t. Similar to the decomposition proposed by [68], we have the following decomposition for this loss:

$$\sigma\Big(\Phi^{T}_{k^*}(t)\cdot\theta-f_{k^*}\big(\Psi_{k^*}(t),y^{*}_{k^*};\pi\big)\Big)-\sigma\Big(\Phi^{T}_{\tilde{k}}(t)\cdot\theta-f_{\tilde{k}}\big(\Psi_{\tilde{k}}(t),y^{*}_{\tilde{k}};\pi\big)\Big) \tag{2.29}$$

$$\leq\sigma\Big(\Phi^{T}_{k^*}(t)\cdot\theta-f_{k^*}\big(\Psi_{k^*}(t),y^{*}_{k^*};\pi\big)\Big)-\sigma\Big(\Phi^{T}_{k^*}(t)\cdot\hat{\theta}(t)-f_{k^*}\big(\Psi_{k^*}(t),y^{*}_{k^*};\hat{\pi}(t)\big)\Big) \tag{2.30}$$

$$+\sigma\Big(\Phi^{T}_{k^*}(t)\cdot\hat{\theta}(t)-f_{k^*}\big(\Psi_{k^*}(t),y^{*}_{k^*};\hat{\pi}(t)\big)\Big)-\sigma\Big(\Phi^{T}_{\tilde{k}}(t)\cdot\hat{\theta}(t)-f_{\tilde{k}}\big(\Psi_{\tilde{k}}(t),y^{*}_{\tilde{k}};\hat{\pi}(t)\big)\Big) \tag{2.31}$$

$$+\sigma\Big(\Phi^{T}_{\tilde{k}}(t)\cdot\hat{\theta}(t)-f_{\tilde{k}}\big(\Psi_{\tilde{k}}(t),y^{*}_{\tilde{k}};\hat{\pi}(t)\big)\Big)-\sigma\Big(\Phi^{T}_{\tilde{k}}(t)\cdot\theta-f_{\tilde{k}}\big(\Psi_{\tilde{k}}(t),y^{*}_{\tilde{k}};\pi\big)\Big). \tag{2.32}$$

In Proposition 1, we derived a high-probability estimation bound on the difference between the true and estimated expected rewards of any treatment k. We denote this high-probability bound on the RHS of (2.5) by $\Lambda^{k}_{t}(\delta)$. Therefore, the first term (2.30) and the last term (2.32) in the above loss decomposition are bounded by $\Lambda^{k^*}_{t}(\delta)$ and $\Lambda^{\tilde{k}}_{t}(\delta)$ for treatments k^* and $\tilde{k}$, respectively. We then bound the second term (2.31) as follows:

$$\begin{aligned}
&\sigma\Big(\Phi^{T}_{k^*}(t)\cdot\hat{\theta}(t)-f_{k^*}\big(\Psi_{k^*}(t),y^{*}_{k^*};\hat{\pi}(t)\big)\Big)-\sigma\Big(\Phi^{T}_{\tilde{k}}(t)\cdot\hat{\theta}(t)-f_{\tilde{k}}\big(\Psi_{\tilde{k}}(t),y^{*}_{\tilde{k}};\hat{\pi}(t)\big)\Big)\\
=\;&\sigma\Big(\Phi^{T}_{k^*}(t)\cdot\hat{\theta}(t)-f_{k^*}\big(\Psi_{k^*}(t),y^{*}_{k^*};\hat{\pi}(t)\big)\Big)+\Lambda^{k^*}_{t}(\delta)\\
&-\sigma\Big(\Phi^{T}_{\tilde{k}}(t)\cdot\hat{\theta}(t)-f_{\tilde{k}}\big(\Psi_{\tilde{k}}(t),y^{*}_{\tilde{k}};\hat{\pi}(t)\big)\Big)-\Lambda^{k^*}_{t}(\delta)\\
\leq\;&\sigma\Big(\Phi^{T}_{\tilde{k}}(t)\cdot\hat{\theta}(t)-f_{\tilde{k}}\big(\Psi_{\tilde{k}}(t),y^{*}_{\tilde{k}};\hat{\pi}(t)\big)\Big)+\Lambda^{\tilde{k}}_{t}(\delta)\\
&-\sigma\Big(\Phi^{T}_{\tilde{k}}(t)\cdot\hat{\theta}(t)-f_{\tilde{k}}\big(\Psi_{\tilde{k}}(t),y^{*}_{\tilde{k}};\hat{\pi}(t)\big)\Big)-\Lambda^{k^*}_{t}(\delta)\\
=\;&\Lambda^{\tilde{k}}_{t}(\delta)-\Lambda^{k^*}_{t}(\delta).
\end{aligned} \tag{2.33}$$

In the above inequality, we deploy the fact that the treatment $\tilde{k}$ selected by the algorithm has the largest upper-confidence bound compared to other treatments. Now, if we plug (2.33) in the second term (2.31), apply Proposition 1 for the terms (2.30) and (2.32), and then sum over all patients arriving over T time steps, we establish the following bound on the *contextual bandit loss*:

$$\sum_{t=1}^{T} \left(\mathbf{V}^*_{k^*}(t) - \mathbf{V}^*_{\tilde{k}}(t) \right) \leq 2 \sum_{t=1}^{T} \Lambda_t^{\tilde{k}}(\delta)\,, \tag{2.34}$$

where

$$\begin{aligned}
&\Lambda_t^{\tilde{k}}(\delta) \\
&= \frac{1}{2c_\sigma} \left\| \Phi_{\tilde{k}}(t) \right\|_{V_t^{-1}} \left(\sqrt{2 \log \left(\frac{\det(V_t)^{1/2} \det(\gamma I)^{-1/2}}{\delta} \right)} + \sqrt{2(d_1 + K) N(t) \log \left(\frac{N(t)}{d_1 + K} \right)} \right. \\
&\left. + \kappa c_\theta \right) + \frac{\max\{\alpha_{\tilde{k}}, \beta_{\tilde{k}}\}}{2} \left\| \Upsilon_{\tilde{k}}(t) \right\|_{U_t^{-1}} \left(\sqrt{2\lambda^2 \log \left(\frac{\det(U_t)^{1/2} \det(\nu I)^{-1/2}}{\delta} \right)} \right. \\
&\left. + c_\psi \sqrt{4(d_2 + 2K) N(t) \log \left(\frac{N(t)}{d_2 + 2K} \right)} + \eta c_\pi \right).
\end{aligned}$$

The following root-squared expression in $\Lambda_t^{\tilde{k}}(\delta)$ can be further simplified by the following algebra:

$$2 \log \left(\frac{\det(V_t)^{1/2}\ \det(\gamma I)^{-1/2}}{\delta} \right) = 2 \log \left(\det(V_t)^{1/2} \right) + 2 \log \left(\frac{\det(\gamma I)^{-1/2}}{\delta} \right),$$

where $V_t = \sum_{s=1}^{t-1} \Phi_{\tilde{k}}(s) \cdot \Phi_{\tilde{k}}^T(s) + \gamma I \in \mathbb{R}^{(d_1+K)\times(d_1+K)}$ is a positive definite matrix. Then, $\text{trace}(V_t)$ is equal to the sum of its eigenvalues, and $\det(V_t)$ is product of its eigenvalues. Thus, by using the inequality of arithmetic and geometric means i.e., $\frac{1}{n} \sum_{i=1}^{n} x_i \geq \sqrt[n]{\prod_{i=1}^{n} x_i}$, and

$\|\Phi_k(t)\| \leq 1$, we have:

$$\det(V_t) = \prod_{i=1}^{d_1+K} \lambda_i \leq \left(\frac{1}{d_1+K}\sum_{i=1}^{d_1+K}\lambda_i\right)^{d_1+K} = \left(\frac{1}{d_1+K}\text{trace}\,(V_t)\right)^{d_1+K}$$
$$= \left(\frac{1}{d_1+K}\left(\sum_{s=1}^{t-1}\|\Phi_{\tilde{k}}(s)\|^2 + \text{trace}\,(\gamma I)\right)\right)^{d_1+K}$$
$$= \left(\frac{1}{d_1+K}\left(\sum_{s=1}^{t-1}\|\Phi_{\tilde{k}}(s)\|^2 + \gamma\text{trace}\,(I)\right)\right)^{d_1+K}$$
$$\leq \left(\frac{1}{d_1+K}\left(t+\gamma(d_1+K)\right)\right)^{d_1+K} = \left(\gamma + \frac{t}{d_1+K}\right)^{d_1+K}, \tag{2.35}$$

where λ_i is the i^{th} eigenvalue of the matrix V_t.

The above implies that $2\log\left(\det(V_t)^{1/2}\right) \leq (d_1+K)\log\left(\gamma + \frac{t}{d_1+K}\right)$, and so we have:

$$2\log\left(\frac{\det(V_t)^{1/2}\det(\gamma I)^{-1/2}}{\delta}\right) = 2\log\left(\det(V_t)^{1/2}\right) + 2\log\left(\frac{\det(\gamma I)^{-1/2}}{\delta}\right)$$
$$\leq (d_1+K)\log\left(\gamma + \frac{t}{d_1+K}\right) + 2\log\left(\det(\gamma I)^{-1/2}\right) + 2\log\left(\frac{1}{\delta}\right)$$
$$\leq (d_1+K)\log\left(1 + \frac{t}{\gamma(d_1+K)}\right) + \log\left(\frac{1}{\delta^2}\right). \tag{2.36}$$

A similar bound can be derived for the third root-squared term in $\Lambda_t^{\tilde{k}}(\delta)$. Furthermore, the summation of norms of feature vectors can be upper bounded using Lemma II.7 (see Appendix C) as:

$$\sum_{t=1}^{T}\|\Phi_{\tilde{k}}(t)\|^2_{V_t^{-1}} \leq 2\log\left(\frac{\det(V_{T+1})}{\det(\gamma I)}\right).$$

Note that in (2.35), we derived $\det(V_{T+1}) \leq \left(\gamma + \frac{T}{d_1+K}\right)^{d_1+K}$, and $\det(\gamma I) = \gamma^{(d+K_1)}\det(I) = \gamma^{(d+K_1)}$. Therefore, we establish the following using Cauchy-Schwartz inequality:

$$\sum_{t=1}^{T}\|\Phi_{\tilde{k}}(t)\|_{V_t^{-1}} \leq \sqrt{T}\sqrt{\sum_{t=1}^{T}\|\Phi_{\tilde{k}}(t)\|^2_{V_t^{-1}}} \leq \sqrt{T}\sqrt{2(d_1+K)\log\left(1+\frac{T}{\gamma(d_1+K)}\right)}. \tag{2.37}$$

A similar bound can be derived for $\sum_{t=1}^{T}\|\Upsilon_{\tilde{k}}(t)\|_{U_t^{-1}}$ as well. Finally, to complete the proof,

if we plug (2.36) and (2.37) in (2.34), we obtain the following bound:

$$
\begin{aligned}
2\sum_{t=1}^{T}\Lambda_t^{\tilde{k}}(\delta) &\leq \frac{1}{c_\sigma}\sum_{t=1}^{T}\|\Phi_{\tilde{k}}(t)\|_{V_t^{-1}}\left(\sqrt{(d_1+K)\log\left(1+\frac{T}{\gamma(d_1+K)}\right)+\log\left(\frac{1}{\delta^2}\right)}\right.\\
&\left.+\sqrt{2(d_1+K)N_{\max}\log\left(\frac{N_{\max}}{d_1+K}\right)}+\kappa c_\theta\right)+\max\{\alpha_k,\beta_k\}\sum_{t=1}^{T}\|\Upsilon_{\tilde{k}}(t)\|_{V_t^{-1}}\\
&\left(\lambda\sqrt{(d_2+2K)\log\left(1+\frac{T}{\nu(d_2+2K)}\right)+\log\left(\frac{1}{\delta^2}\right)}\right.\\
&\left.+c_\psi\sqrt{4(d_2+2K)N_{\max}\log\left(\frac{N_{\max}}{d_2+2K}\right)}+\eta c_\pi\right)\\
&\leq \frac{1}{c_\sigma}\sqrt{2T(d_1+K)\log\left(1+\frac{T}{\gamma(d_1+K)}\right)}\left(\sqrt{(d_1+K)\log\left(1+\frac{T}{\gamma(d_1+K)}\right)+\log\left(\frac{1}{\delta^2}\right)}\right.\\
&\left.+\sqrt{2(d_1+K)N_{\max}\log\left(\frac{N_{\max}}{d_1+K}\right)}+\kappa c_\theta\right)\\
&+\max\{\alpha_k,\beta_k\}\sqrt{2T(d_2+2K)\log\left(1+\frac{T}{\nu(d_2+2K)}\right)}\\
&\left(\lambda\sqrt{(d_2+2K)\log\left(1+\frac{T}{\nu(d_2+2K)}\right)+\log\left(\frac{1}{\delta^2}\right)}\right.\\
&\left.+c_\psi\sqrt{4(d_2+2K)N_{\max}\log\left(\frac{N_{\max}}{d_2+2K}\right)}+\eta c_\pi\right)\\
&= L(\delta,T)+Q(\delta,T),
\end{aligned}
$$

where $L(T,\delta)$ and $Q(T,\delta)$ are defined in Proposition 2.

In the above derivation, $N_{\max}=\max_{1\leq t\leq T}N(t)$ is the maximum number of unrealized feedbacks by the time period T. Applying Lemma II.6 (see Appendix C), $N_{\max}$ is upper bounded by $2\mu_D+\tilde{\sigma}\Big(\sqrt{2\log T}+\sqrt{2\log\left(1/\delta\right)+c'(\tilde{\sigma}\sqrt{2\log T}+1)}+c\Big)$, with probability $1-\delta$, where $c=2\,\tilde{\sigma}^2\log\left(2\sigma_D^2+1\right)+1$, $c'=2\log(2\sigma_D^2+1)$ and $\tilde{\sigma}=\sigma_D\sqrt{p+2}$.

Plugging the above bound in (2.34), the following high-probability bound is obtained on the contextual bandit loss with probability $1-2\delta$:

$$
\sum_{t=1}^{T}\Big(\mathbf{V}_{k^*}^*(t)-\mathbf{V}_{\tilde{k}}^*(t)\Big)\leq L(\delta,T)+Q(\delta,T).
$$

Part II (S-SGD sub-optimality loss): Similar to our procedure that we developed in Part

II of the proof of Theorem II.1, the S-SGD sub-optimality loss is bounded with probability $1-5\delta$ as:

$$\sum_{t=1}^{T}\left(\mathbf{V}^{*}_{\tilde{k}}(t)-\hat{\mathbf{V}}_{\tilde{k}}(t)\right)\leq P(\delta,T)\sqrt{T}+\frac{L(T,\delta)}{2}+Q(T,\delta),$$

where $P(\delta,T)=\frac{1}{4}G\sqrt{2\rho\log(1/\delta)}+\frac{3}{4}\xi G\sqrt{1+\mu_D+\sqrt[p+1]{\sigma_D^2\log(T/\delta)}}$.

Part III (Estimation loss): We bound the estimation loss $\sum_{t=1}^{T}\left(\hat{\mathbf{V}}_{\tilde{k}}(t)-\mathbf{V}_{\tilde{k}}(t)\right)$ with probability at least $1-2\delta$ using Proposition 1 and applying a similar argument that we developed in Part I of the proof of Corollary II.2 (see Appendix B) as follows:

$$\sum_{t=1}^{T}\left(\hat{\mathbf{V}}_{\tilde{k}}(t)-\mathbf{V}_{\tilde{k}}(t)\right)\leq\frac{L(T,\delta)+Q(T,\delta)}{2}.$$

Finally, putting all the three established high-probability bounds together, the following bound holds on the regret:

$$\begin{aligned}
&\textbf{Regret}(T,\theta,\pi)\leq P(\delta,T)\sqrt{T}+2L(T,\delta)+\frac{5Q(T,\delta)}{2}\\
&=\frac{G}{4}\left(\sqrt{2\rho\log(1/\delta)}+3\xi\sqrt{1+\mu_D+\sqrt[p+1]{\sigma_D^2\log(T/\delta)}}\right)\sqrt{T}\\
&+\frac{2}{c_\sigma}\sqrt{2T(d_1+K)\log\left(1+\frac{T}{\gamma(d_1+K)}\right)}\left(\sqrt{(d_1+K)\log\left(1+\frac{T}{\gamma(d_1+K)}\right)+\log\left(\frac{1}{\delta^2}\right)}\right.\\
&\left.+\sqrt{2(d_1+K)N_{\max}\log\left(\frac{N_{\max}}{d_1+K}\right)}+\kappa c_\theta\right)\\
&+\frac{5\max\{\alpha_k,\beta_k\}}{2}\sqrt{2T(d_2+2K)\log\left(1+\frac{T}{\nu(d_2+2K)}\right)}\\
&\left(\lambda\sqrt{(d_2+2K)\log\left(1+\frac{T}{\nu(d_2+2K)}\right)+\log\left(\frac{1}{\delta^2}\right)}\right.\\
&\left.+c_\psi\sqrt{4(d_2+2K)N_{\max}\log\left(\frac{N_{\max}}{d_2+2K}\right)}+\eta c_\pi\right)\\
&=\mathcal{O}\Bigg(\sqrt{T}\,(d_1+K)\left(\log\left(T/(d_1+K)\right)+\sqrt{N_{\max}\log(N_{\max}/(d_1+K))}\right)+\\
&\qquad\sqrt{T}\,(d_2+2K)\left(\log\left(T/(d_2+2K)\right)+\sqrt{N_{\max}\log(N_{\max}/(d_2+2K))}\right)\Bigg),
\end{aligned}$$

which completes the proof when we set $\delta = 1/T$. $\square$

Lemma II.3 (**Sub-Gaussian Stochastic Sub-gradient**). For the penalty function (2.2), there exists a positive constant ξ such that we have the following:

1. The stochastic sub-gradient $\widetilde{\nabla} f_k$ defined in Step (5b) is ρ-sub-Gaussian.
2. The second moment of the stochastic sub-gradient $\widetilde{\nabla} f_k$ is bounded by ξ^2, i.e.,

$$\mathbb{E}\Big[\widetilde{\nabla} f_k^2\Big] \leq \xi^2 \text{ where } \xi = \max(\alpha_k, -\beta_k)\sqrt{\mathbb{E}\Big[[\widetilde{\tau}]_k^2\Big]}.$$

Proof. Proof of Lemma II.3: Recall that we use the following patient-specific dosage penalty function:

$$f_k\big(\Psi_k, y_k; \pi\big) = \alpha_k\Big(\Psi_k^T \cdot \widetilde{\omega} + y_k^T \cdot \widetilde{\tau} - q\Big)^+ + \beta_k\Big(q - \Psi_k^T \cdot \widetilde{\omega} - y_k^T \cdot \widetilde{\tau}\Big)^+,$$

and its sub-gradient with respect to the k^{th} element of y_k is calculated as follows:

$$\widetilde{\nabla} f_k = [\widetilde{\tau}]_k\Big[\alpha_k \cdot \mathbb{1}\Big(\Psi_k^T \cdot \widetilde{\omega} + y_k^T \cdot \widetilde{\tau} > q\Big) - \beta_k \cdot \mathbb{1}\Big(\Psi_k^T \cdot \widetilde{\omega} + y_k^T \cdot \widetilde{\tau} < q\Big)\Big].$$

Part I: We shall prove that the sub-gradient $\widetilde{\nabla} f$ is ρ-sub-Gaussian. Using Taylor's series expansion, we have the following:

$$\mathbb{E}\Big[\exp\big(s\ \widetilde{\nabla} f_k\big)\Big] = 1 + s\ \mathbb{E}\Big[|\widetilde{\nabla} f_k|\Big] + \frac{s^2\,\mathbb{E}\Big[|\widetilde{\nabla} f_k|^2\Big]}{2!} + \sum_{m=3}^{\infty} \frac{s^m\,\mathbb{E}\Big[|\widetilde{\nabla} f_k|^m\Big]}{m!},$$

which implies that we have:

$$\frac{\sum\limits_{m=3}^{\infty} \dfrac{s^m\ \mathbb{E}\Big[|\widetilde{\nabla} f_k|^m\Big]}{m!}}{s^2} = \frac{\mathbb{E}\Big[\exp\big(s\ \widetilde{\nabla} f_k\big)\Big] - s\ \mathbb{E}\Big[|\widetilde{\nabla} f_k|\Big] - \dfrac{s^2\ \mathbb{E}\Big[\widetilde{\nabla} f_k^2\Big]}{2} - 1}{s^2},$$

which converges to 0 as $s \to 0$ by using L'Hôspital's rule. This means that:

$$\frac{\sum\limits_{m=3}^{\infty} \dfrac{s^m\ \mathbb{E}\Big[|\widetilde{\nabla} f_k|^m\Big]}{m!}}{s^2} = o(s^2).$$

Moreover, we note from the sub-gradient that $|\widetilde{\nabla} f_k| \leq \max(\alpha_k, -\beta_k)\ [\widetilde{\tau}]_k$. Now consider:

$$\begin{aligned}
\mathbb{E}\Big[\exp\left(s\ \widetilde{\nabla} f_k\right)\Big] &\leq\ 1 + \sum_{m=2}^{\infty} \frac{s^m\, \mathbb{E}\Big[|\widetilde{\nabla} f_k|^m\Big]}{m!} \\
&=\ 1 + \frac{s^2\, \mathbb{E}\Big[|\widetilde{\nabla} f_k|^2\Big]}{2!} + \sum_{m=3}^{\infty} \frac{s^m\, \mathbb{E}\Big[|\widetilde{\nabla} f_k|^m\Big]}{m!} \\
&\leq\ 1 + \frac{s^2\ (\max(\alpha_k, -\beta_k))^2}{2!}\ \mathbb{E}\Big[[\widetilde{\tau}]_k^2\Big] + \sum_{m=3}^{\infty} \frac{s^m\ (\max(\alpha_k, -\beta_k))^m}{m!}\ \mathbb{E}\Big[[\widetilde{\tau}]_k^m\Big] \\
&\leq\ 1 + \frac{s^2\ (\max(\alpha_k, -\beta_k))^2}{2!}\ \mathbb{E}\Big[[\widetilde{\tau}]_k^2\Big] + o(s^2).
\end{aligned}$$

On the other hand, we know that for any $\rho \geq 0$, we have the following:

$$\exp\left(\frac{s^2\,\rho^2}{2}\right) = 1 + \frac{s^2\,\rho^2}{2} + o(s^2).$$

Therefore, if we choose ρ such that $\rho^2 > (\max(\alpha_k, -\beta_k))^2\, \mathbb{E}\Big[[\widetilde{\tau}]_k^2\Big]$, then for all $s > 0$, we have:

$$\mathbb{E}\Big[\exp\left(s\ \widetilde{\nabla} f_k\right)\Big]\ \leq\ \exp\left(\frac{s^2\,\rho^2}{2}\right),$$

which shows that sub-gradient $\widetilde{\nabla} f_k$ is a ρ-sub-Gaussian random variable (see Definition 1 in §2.2).

Part II: We also show that the sub-gradient $\widetilde{\nabla} f$ has a bounded second moment. Since $|\widetilde{\nabla} f_k| \leq \max(\alpha_k, -\beta_k)\ [\widetilde{\tau}]_k$, then we have $\mathbb{E}\Big[\widetilde{\nabla} f_k^2\Big]\ \leq\ \xi^2$ where $\xi = \max(\alpha_k, -\beta_k)\sqrt{\mathbb{E}\Big[[\widetilde{\tau}]_k^2\Big]}$. □

Proposition 5 (**High-Probability Regret Bound for Online Sub-Gaussian B-SGD**). Let $\{y_i\}_{i=1}^n \in \mathbb{R}^d$ be a sequence obtained by the projected online B-SGD algorithm under stochastic delay with respect to convex and L-Lipchitz functions $f(.)$ with a domain $\mathcal{K}$, i.e.,

$$y_1 \in \mathcal{K},\ y_{i+1} = \mathbf{Proj}_{\mathcal{K}_\vartheta}\Big(y_i - \eta \sum_{s \in \mathcal{S}_i} \widetilde{g}_s\Big),\ \ \forall i = 1, ..., n-1,$$

where $\widetilde{g}_s = \frac{d}{\vartheta}\ f_s(y_s + \vartheta u_s)\ u_s$ is an approximate stochastic gradient of f_s at y_s, u_s is a random unit vector sampled from the Euclidean sphere $\mathbb{S} = \{u \in \mathbb{R}^d \mid \|u\| = 1\}$, $\vartheta > 0$ is a perturbation parameter, η is step size at each iteration, $\mathcal{S}_i = \{s \in [n] \mid s + D(s) - 1 = i\}$ is the set of iterations whose feedback appears at the end of iteration i, and $D(s)$ is stochastic delay. We make the following assumptions:

1. Diameter of the set $\mathcal{K}$ is bounded by a constant G, i.e., $\sup_{y_1,y_2\in\mathcal{K}} \|y_1 - y_2\| \leq G$. The absolute value of the function at any point is bounded by a constant C, i.e., $\sup_{y\in\mathcal{K}} |f(y)| \leq C$.

2. The set $\mathcal{K}$ contains the Euclidean ball $\mathbb{B} = \{y \in \mathbb{R}^d \mid \|y\| \leq 1\}$ centered at the zero vector.

3. The set $\mathcal{K}_\vartheta$ is the Minkowski set corresponding to the set $\mathcal{K}$, defined by $\mathcal{K}_\vartheta = \{y \mid \frac{1}{(1-\vartheta)} y \in \mathcal{K}\}$.

4. The stochastic delay $D(s)$ satisfies the regularity condition (2.4), and $D = \sum_{i=1}^n D(i)$.

If we choose the step sizes $\eta = \frac{G}{d(n+D)^{3/4}}$, the following bound holds with probability at least $1 - 2\delta$:

$$\sum_{i=1}^{n} \Big(\mathbb{E}\big[f_i(y_i + \vartheta u_i)\big] - f_i(y^*) \Big) \leq \frac{dG}{2} \Big(1 + 5C^2 + \frac{8L}{d} \Big) \left(1 + \mu_D + \sqrt[p+1]{\sigma_D^2 \log(n/\delta)} \right)^{3/4} n^{3/4} + G \sqrt{2dC \log(1/\delta) \left(1 + \mu_D + \sqrt[p+1]{\sigma_D^2 \log(n/\delta)} \right)^{3/4} n^{3/4}},$$

where $y^* = \arg\min_{y\in\mathcal{K}} \sum_{i=1}^n f_i(y)$.

Proof. Proof of Proposition 5: First, note that [69] propose the following:

$$\widetilde{g} = \frac{d}{\vartheta} f(y + \vartheta u)\, u$$

as a stochastic gradient estimator of f at y, where $\vartheta > 0$ is a perturbation parameter, and u is a random unit vector randomly selected from the Euclidean sphere $\mathbb{S}$. Stoke's theorem shows that $\widetilde{g}$ is an unbiased gradient estimator of the ϑ-*smoothed* version $\hat{f}(y) = \mathbb{E}_{\mathrm{v}\in\mathbb{B}}\big[f(y + \vartheta \mathrm{v})\big]$ of any convex (not necessarily differentiable) function f, where v is randomly selected from the Euclidean ball $\mathbb{B} = \{\mathrm{v} \in \mathbb{R}^d \mid \|\mathrm{v}\| \leq 1\}$ (see Lemma 2.1 of [69]). This implies for $\vartheta > 0$:

$$\mathbb{E}_{u\in\mathbb{S}}\Big[f(y + \vartheta u) u \;\Big|\; y \Big] = \frac{\vartheta}{d} \nabla \hat{f}(y) = \frac{\vartheta}{d} \nabla \mathbb{E}_{\mathrm{v}\in\mathbb{B}}\big[f(y + \vartheta \mathrm{v})\big],$$

where $\mathbb{S} = \{u \in \mathbb{R}^d \mid \|u\| = 1\}$ is the Euclidean sphere centered at the zero vector.

Given that the value of the function f is bounded by a constant C at any point, we have:

$$\|\widetilde{g}\| = \left\| \frac{d}{\vartheta} f(y + \vartheta u)\, u \right\| \leq \frac{d}{\vartheta} C.$$

Using a similar argument as we developed in Lemma II.3, we can prove that the gradient estimator $\widetilde{g}$ is ρ-sub-Gaussian and its second moment is bounded by $\mathbb{E}\big[\,\|\widetilde{g}\|^2\,\big] \leq \rho^2$, where $\rho = \frac{d}{\vartheta}\,C$. Also, the ϑ-*smoothed* version $\hat{f}$ is a good approximation of f, because for $y \in \mathcal{K}$:

$$\begin{aligned}\Big|\hat{f}(y) - f(y)\Big| &= \Big|\ \mathbb{E}_{\mathrm{v}\in\mathbb{B}}\Big[f(y+\vartheta \mathrm{v})\Big] - f(y)\ \Big| \\ &\leq\ \mathbb{E}_{\mathrm{v}\in\mathbb{B}}\big[\,|f(y+\vartheta \mathrm{v}) - f\,(y)|\,\big] \\ &\leq\ L\vartheta\ \mathbb{E}_{\mathrm{v}\in\mathbb{B}}\big[\,\|\mathrm{v}\|\,\big] \leq L\vartheta, \end{aligned} \tag{2.38}$$

where the equality is by definition of $\hat{f}$, the first inequality is by Jensen's inequality, the second inequality is due to f being L-Lipschitz, and the last inequality holds because $\mathrm{v} \in \mathbb{B}$.

Moreover, note that we project onto the *shrunken* set $\mathcal{K}_\vartheta = \{y\ |\frac{1}{(1-\vartheta)}y \in \mathcal{K}\}$ instead of the original set $\mathcal{K}$ to avoid moving outside of the set $\mathcal{K}$ when we add the random sampling from the Euclidean sphere $\mathbb{S}$. For these two sets $\mathcal{K}$ and $\mathcal{K}_\vartheta$, we have the following properties:

- **Property 1**: The shrunken set $\mathcal{K}_\vartheta$ is convex for any $0 < \vartheta < 1$.
- **Property 2**: $\forall y \in \mathcal{K}_\vartheta$, $\mathbb{B}_\vartheta(y) = \{x \mid x = y + \vartheta u\} \subseteq \mathcal{K}$, i.e., all balls of radius ϑ around points $y \in \mathcal{K}_\vartheta$ are contained in $\mathcal{K}$, because $\mathcal{K}$ is convex and $u\mathbb{B} \subseteq \mathcal{K}$, so $\mathcal{K}_\vartheta + \vartheta u\mathbb{B} \subseteq (1-\vartheta)\mathcal{K} + \vartheta\mathcal{K} = \mathcal{K}$.
- **Property 3**: $\forall y \in \mathcal{K}$, $\exists y_\vartheta \in \mathcal{K}_\vartheta$ such that $\|y_\vartheta - y\| \leq \vartheta G$, where G is the diameter of $\mathcal{K}$.

Now, let $y^* = \arg\min_{y\in\mathcal{K}} \sum_{i=1}^n f_i(y)$, and $y_\vartheta^* = \mathbf{Proj}_{\mathcal{K}_\vartheta}(y^*)$. Denote $\hat{f}_i(y_i) = \mathbb{E}_{\mathrm{v}_i\in\mathbb{B}}\big[f_i(y_i + \vartheta \mathrm{v}_i)\big]$ for shorthand, where $\hat{f}_i(y_i)$ is the ϑ-*smoothed* version of $f_i(y_i)$ at y_i. Also, let $x_i = y_i + \vartheta u_i$. We can then bound the regret of the B-SGD as follows:

$$\begin{aligned}\sum_{i=1}^n \Big(\mathbb{E}\big[f_i(x_i)\big] - f_i(y^*)\Big) &\leq\ \sum_{i=1}^n \Big(\mathbb{E}\big[f_i(x_i)\big] - f_i(y_\vartheta^*)\Big) + \vartheta nLG \\ &\leq\ \sum_{i=1}^n \mathbb{E}\Big[f_i\,(y_i)\,\Big] - \sum_{i=1}^n f_i\,(y_\vartheta^*) + 2\vartheta nLG \\ &\leq\ \sum_{i=1}^n \mathbb{E}\Big[\hat{f}_i\,(y_i)\,\Big] - \sum_{i=1}^n \hat{f}_i\,(y_\vartheta^*) + 4\vartheta nLG. \end{aligned} \tag{2.39}$$

The first inequality is by Property 3 and f being L-Lipschitz, the second inequality is established by $|f_i\,(x_i) - f_i\,(y_i)\,| \leq L\,\|x_i - y_i\| \leq \vartheta L$. Also, note that since the set $\mathcal{K}$ contains the Euclidean ball, the diameter G of this set is greater than 1 (so $2\vartheta nL \leq 2\vartheta nLG$). The third inequality is by (2.38).

In (2.39), $\sum_{i=1}^{n}\mathbb{E}\big[\hat{f}_i\left(y_i\right)\big]-\sum_{i=1}^{n}\hat{f}_i\left(y_\vartheta^*\right)$ is indeed equal to the high-probability regret of implementing the high-probability S-SGD algorithm on the functions $\hat{f}$ and over $\mathcal{K}_\vartheta$ under stochastic delay. In Proposition 3, we develop the following high-probability bound with probability $1-\delta$:

$$\sum_{i=1}^{n}\mathbb{E}\Big[\hat{f}_i\left(y_i\right)\Big]-\sum_{i=1}^{n}\hat{f}_i\left(y_\vartheta^*\right)\leq\frac{G^2}{2\eta}+\frac{\eta}{2}n\xi^2+2\xi^2\eta D+\sqrt{2G^2\rho\log(1/\delta)n},\tag{2.40}$$

where $\xi=\rho=\frac{d}{\vartheta}\,C$. Plugging the high-probability bound (2.40) into (2.39), we can establish the following high-probability bound with probability $1-\delta$:

$$\begin{aligned}\sum_{i=1}^{n}\Big(\mathbb{E}\big[f_i(x_i)\big]-f_i(y^*)\Big)&\leq\sum_{i=1}^{n}\mathbb{E}\Big[\hat{f}_i\left(y_i\right)\Big]-\sum_{i=1}^{n}\hat{f}_i\left(y_\vartheta^*\right)+4\vartheta nLG\\&\leq\frac{G^2}{2\eta}+\frac{\eta n}{2}\left(\frac{dC}{\vartheta}\right)^2+2\eta D\left(\frac{dC}{\vartheta}\right)^2+4\vartheta nLG\\&\quad+G\sqrt{2\left(\frac{dC}{\vartheta}\right)\log(1/\delta)n}\\&\leq\frac{dG}{2}\Big((n+D)^{3/4}+\frac{C^2n}{(n+D)^{1/4}}+\frac{4C^2D}{(n+D)^{1/4}}+\frac{8Ln}{d(n+D)^{1/4}}\Big)\\&\quad+G\sqrt{2dC\log(1/\delta)(n+D)^{1/4}n}\\&\leq\frac{dG}{2}\Big(1+5C^2+\frac{8L}{d}\Big)(n+D)^{3/4}+G\sqrt{2dC\log(1/\delta)(n+D)^{3/4}},\end{aligned}$$

where we set $\eta=\frac{G}{d(n+D)^{3/4}}$ and $\vartheta=\frac{1}{(n+D)^{1/4}}$ in the second inequality. To get the last inequality, we replace both n and D with $(n+D)$ in the third inequality.

In Proposition 3, we establish a high-probability bound on the total delay $D=\sum_{i=1}^{n}D(i)$ under (2.4) by $D\leq n(\mu_D+\sqrt[p+1]{\sigma_D^2\log(n/\delta)})$ with probability $1-\delta$. Thus, the high-probability regret bound, which holds with probability $1-2\delta$, is as follows:

$$\begin{aligned}\sum_{i=1}^{n}\Big(\mathbb{E}\big[f_i(x_i)\big]-f_i(y^*)\Big)&\leq\frac{dG}{2}\Big(1+5C^2+\frac{8L}{d}\Big)(n+D)^{3/4}+G\sqrt{2dC\log(1/\delta)(n+D)^{3/4}}\\&\leq\frac{dG}{2}\Big(1+5C^2+\frac{8L}{d}\Big)\left(1+\mu_D+\sqrt[p+1]{\sigma_D^2\log(n/\delta)}\right)^{3/4}n^{3/4}\\&\quad+G\sqrt{2dC\log(1/\delta)\left(1+\mu_D+\sqrt[p+1]{\sigma_D^2\log(n/\delta)}\right)^{3/4}n^{3/4}},\end{aligned}$$

It is worth noting that when there is *no delay* in observing feedbacks, the above high-

probability regret becomes the following with probability $1-2\delta$:

$$\begin{aligned}\sum_{i=1}^{n}\Big(\mathbb{E}\big[f_i(x_i)\big]-f_i(y^*)\Big) &\leq \sum_{i=1}^{n}\mathbb{E}\Big[\hat{f}_i\left(y_i\right)\Big]-\sum_{i=1}^{n}\hat{f}_i\left(y_\vartheta^*\right)+4\vartheta nLG\\ &\leq \frac{G^2}{2\eta}+\frac{\eta}{2}n\xi^2+\sqrt{2G^2\rho\log(1/\delta)n}+4\vartheta nLG\\ &\leq \frac{G^2}{2\eta}+\frac{\eta n}{2}\left(\frac{dC}{\vartheta}\right)^2+G\sqrt{2\left(\frac{dC}{\vartheta}\right)\log(1/\delta)n}+4\vartheta nLG\\ &\leq \frac{dG}{2}\Big(1+C^2+\frac{8L}{d}\Big)n^{3/4}+G\sqrt{2dC\log(1/\delta)n^{3/4}},\end{aligned}$$

where we set $\eta=\frac{G}{dn^{3/4}}$, and $\vartheta=\frac{1}{n^{3/4}}$ in the third inequality. □

Proof. Proof of Proposition 4: Recall the Bayesian regret decomposition in §2.4.3. We bound each part of it for the TS-based B-SGD Bandit algorithm (Algorithm 3 in Appendix E) below.

Part I (Contextual bandit loss): To bound the contextual bandit loss, we follow our approach in the Part I of the proof of Theorem II.1 where we used Proposition 2. Accordingly, the following bound for the contextual bandit loss holds with probability $1-\delta$:

$$\begin{aligned}&\mathbb{E}\Big[\sum_{t=1}^{T}\big(\mathbf{V}_{k^*}^*(t)-\mathbf{V}_k^*(t)\big)\Big]\leq\frac{1}{c_\sigma}\sqrt{2T(d_1+K)\log\left(1+\frac{T}{\gamma(d_1+K)}\right)}\\ &\left(\sqrt{(d_1+K)\log\left(1+\frac{T}{\gamma(d_1+K)}\right)+\log\left(\frac{1}{\delta^2}\right)}\right.\\ &\left.+\sqrt{2(d_1+K)N_{\max}\log\left(\frac{N_{\max}}{d_1+K}\right)+\kappa c_\theta}\right)+T\,\delta=L(T,\delta)+T\delta.\end{aligned}$$

It is worth noting that since the structure of the penalty function f is unknown in §2.4.7, there is no need to learn the parameter π using an online linear regression (that we had in Theorem II.1). Therefore, we only need to include the loss we incurred due to learning the parameter θ in the above bound using an online logistic regression. In particular, we use the high-probability bound that we developed in Part I of the proof of Proposition 1.

Part II (B-SGD sub-optimality loss): Since we use the B-SGD procedure instead of the S-SGD in our algorithm (Algorithm 3), we bound the second term $\mathbb{E}\big[\sum_{t=1}^{T}\big(\mathbf{V}_k^*(t)-\hat{\mathbf{V}}_k(t)\big)\big]$ as follows.

To this aim, we first decompose it using the Lipschitz property of the logistic function,

which is Lipschitz with constant 1/4:

$$\begin{aligned}
&\mathbb{E}\Big[\sum_{t=1}^{T}\big(\mathbf{V}_k^*(t)-\hat{\mathbf{V}}_k(t)\big)\Big]\\
&=\mathbb{E}\Big[\sum_{t=1}^{T}\sigma\left(\Phi_k^T(t)\cdot\theta-f_k\left(\Psi_k(t),y_k^*\right)\right)-\sigma\big(\Phi_k^T(t)\cdot\hat{\theta}(t)-f_k\left(\Psi_k(t),y_k(t)\right)\big)\Big]\\
&\leq\frac{1}{4}\Big(\underbrace{\mathbb{E}\Big[\sum_{t=1}^{T}\Big(f_k\big(\Psi_k(t),y_k(t)\big)-f_k\big(\Psi_k(t),y_k^*\big)\Big)\Big]}_{\text{Part I}}+\underbrace{\mathbb{E}\Big[\sum_{t=1}^{T}\Big(\Phi_k^T(t)\cdot\big(\theta-\hat{\theta}(t)\big)\Big)\Big]}_{\text{Part II}}\Big). \qquad (2.41)
\end{aligned}$$

First, we can establish the following high-probability bound for Part I that holds with probability $1-2\delta$ for the general dosage penalty function that we developed in Proposition 5:

$$\mathbb{E}\Big[\sum_{t=1}^{T}\Big(f_k\left(\Psi_k(t),y_k(t)\right)-f_k\left(\Psi_k(t),y_k^*\right)\Big)\Big]\leq S(\delta,T)\,T^{3/4},$$

where $S(\delta,T)=\frac{G}{4}\left(\frac{1+5C^2+8L}{2}\left(1+\mu_D+\sqrt[p+1]{\sigma_D^2\log(T/\delta)}\right)^{3/4}+\sqrt{2C\log(1/\delta)\left(1+\mu_D+\sqrt[p+1]{\sigma_D^2\log(T/\delta)}\right)^{3/4}1/T^{3/4}}\right)$, and also $y_k^*=\arg\min_{y_k\in[\Delta_k^{LB},\Delta_k^{UB}]}\sum_{t=1}^{T}f_k\left(\Psi_k(t),y_k\right)$.

Second, we use Lemma II.4 (see Appendix C) as well as Part I of the proof of Proposition 1 to bound Part II of (2.41). Hence, it is bounded with probability at least $1-\delta$ as follows:

$$\begin{aligned}
&\mathbb{E}\Big[\sum_{t=1}^{T}\Big(\Phi_k^T(t)\cdot\big(\theta-\hat{\theta}(t)\big)\Big)\Big]\leq\frac{2}{c_\sigma}\sum_{t=1}^{T}\|\Phi_k(t)\|_{V_t^{-1}}\left\|h_t(\theta)-h_t(\bar{\theta}(t))\right\|_{V_t^{-1}}\\
&\leq\frac{2}{c_\sigma}\sqrt{2T(d_1+K)\log\left(1+\frac{T}{\gamma(d_1+K)}\right)}\\
&\left(\sqrt{(d_1+K)\log\left(1+\frac{T}{\gamma(d_1+K)}\right)+\log\left(\frac{1}{\delta^2}\right)}+\sqrt{2(d_1+K)N_{\max}\log\left(\frac{N_{\max}}{d_1+K}\right)}+\kappa c_\theta\right)\\
&=2L(\delta,T),
\end{aligned}$$

where the second inequality is by a similar argument made in Part I of the proof of Corollary II.2.

Inserting the above two bounds in (2.41), the following is obtained with probability at

least $1-3\delta$:

$$\mathbb{E}\Big[\sum_{t=1}^{T}\big(\mathbf{V}_k^*(t)-\hat{\mathbf{V}}_k(t)\big)\Big] \leq S(\delta,T)\,T^{3/4}+\frac{L(\delta,T)}{2}.$$

It is worth noting that in above derivation we convert the high-probability S-SGD bound that we developed in Proposition 5 on the general dosage penalty function into a corresponding high-probability bound on the expected reward.

Part III (Estimation loss): We bound the total estimation loss of $\mathbb{E}\big[\sum_{t=1}^{T}\big(\hat{\mathbf{V}}_k(t)-\mathbf{V}_k(t)\big)\big]$ with probability at least $1-\delta$ using Proposition 1 as follows:

$$\mathbb{E}\Big[\sum_{t=1}^{T}\big(\hat{\mathbf{V}}_k(t)-\mathbf{V}_k(t)\big)\Big] \leq \frac{1}{2c_\sigma}\sqrt{2T(d_1+K)\log\left(1+\frac{T}{\gamma(d_1+K)}\right)}$$
$$\left(\sqrt{(d_1+K)\log\left(1+\frac{T}{\gamma(d_1+K)}\right)+\log\left(\frac{1}{\delta^2}\right)}+\sqrt{2(d_1+K)N_{\max}\log\left(\frac{N_{\max}}{d_1+K}\right)}+\kappa c_\theta\right)$$
$$=\frac{L(\delta,T)}{2}.$$

Note that here there is no parameter π and thus, we do not have any estimation loss for that.

Finally, putting all the above bounds developed in Parts I, II, and II together, we can establish the following bound on the Bayesian regret of the TS-based B-SGD Bandit algorithm:

$$\textbf{BayesianRegret}(T) \leq S(\delta,T)\,T^{3/4}+2L(\delta,T)+T\delta=\widetilde{\mathcal{O}}\Big(T^{3/4}\,\big(\sqrt{1+\mu_D}\big)^{3/4}\Big),$$

which completes the proof where we set $\delta=1/T$. □

2.7.3 Appendix C: Known Results

In this Appendix, we provide some key known results and definitions from the related literature. For completeness, we provide the readers with self-contained and more expository versions of their original proofs and results.

Lemma II.4 (**Initial Upper-Bound on Expected Reward**). For any time period t, the following upper bound on the difference between the true and estimated expected rewards

holds (Adopted from Proposition 1 in [68]):

$$\Big|\sigma\Big(\Phi_k^T(t)\cdot\theta - f_k\big(\Psi_k(t), y_k(t);\pi\big)\Big) - \sigma\Big(\Phi_k^T(t)\cdot\hat{\theta}(t) - f_k\big(\Psi_k(t), y_k(t);\pi\big)\Big)\Big| \\ \leq \frac{1}{2c_\sigma}\,\|\Phi_k(t)\|_{V_t^{-1}}\,\big\|h_t(\theta) - h_t(\bar{\theta}(t))\big\|_{V_t^{-1}}\,.$$

Proof. Proof of Lemma II.4: Let $\mathcal{S}\subset\mathbb{R}^n$ be an open set and also $\omega_1,\omega_2\in\mathbb{R}^n$. Consider a vector-valued function $F:\mathcal{S}\to\mathbb{R}^n$, and assume that it is continuously differentiable. According to the *mean-value theorem* for vector-valued functions, there exist $\bar{\omega}=\lambda\;\omega_1+(1-\lambda)\;\omega_2$, where $0<\lambda<1$ such that:

$$F(\omega_2)-F(\omega_1)=\Big(\int_0^1\nabla F(\bar{\omega})\;d\lambda\Big)\;(\omega_2-\omega_1).$$

To use the mean-value theorem for the following continuously differentiable vector-valued function,

$$h_t(\theta)=\sum_{s=1}^{t-1}\sigma\big(\Phi_k^T(s)\cdot\theta - f_k\big(\Psi_k(s),y_k(s);\pi\big)\big)\;\Phi_k(s)+\kappa\theta,$$

let $\theta^0=\lambda\,\theta+(1-\lambda)\,\hat{\theta}(t)$ with $0<\lambda<1$. Then, we have the following gradient vector:

$$\nabla h_t(\theta^0)=\sum_{s=1}^{t-1}\nabla_{\theta^0}\sigma\big(\Phi_k^T(s)\cdot\theta^0 - f_k\big(\Psi_k(s),y_k(s);\;\pi\big)\big)\;\Phi_k(s)\cdot\Phi_k^T(s)+\kappa I.$$

Let $H_t(\theta^0)=\int_0^1\nabla h_t(\theta^0)\;d\lambda$. Recall $c_\sigma=\inf\limits_{\theta,\pi,\Phi_k(s),\Psi_k(s),y_k(s)}\nabla\sigma\big(\Phi_k^T(s)\cdot\theta-f_k\big(\Psi_k(s),y_k(s);\pi\big)\big)>0$, and $\gamma=\kappa/c_\sigma\geq 1$. This implies $H_t(\theta^0)\succeq c_\sigma\,V_t\succ 0$, where $V_t=\sum_{s=1}^{t-1}\Phi_k(s)\cdot\Phi_k^T(s)+\gamma I$ is the design matrix corresponding to the first $t-1$ time-steps of the observed features Φ_k. Therefore, $H_t(\theta^0)$ is a positive definite and non-singular matrix. According to the mean-value theorem, we then have:

$$h_t(\theta)-h_t(\hat{\theta}(t))=H_t(\theta^0)\cdot\Big(\theta-\hat{\theta}(t)\Big)\Rightarrow\Big(\theta-\hat{\theta}(t)\Big)=H_t^{-1}(\theta^0)\cdot\Big(h_t(\theta)-h_t(\hat{\theta}(t))\Big)\,.$$

Therefore, we can derive the following bound for each time period t:

$$\begin{aligned}
&\Big|\sigma\Big(\Phi_k^T(t)\cdot\theta - f_k\big(\Psi_k(t), y_k(t);\pi\big)\Big) - \sigma\Big(\Phi_k^T(t)\cdot\hat{\theta}(t) - f_k\big(\Psi_k(t), y_k(t);\pi\big)\Big)\Big| \\
&\leq \frac{1}{4}\left|\Phi_k^T(t)\cdot\theta - \Phi_k^T(t)\cdot\hat{\theta}(t)\right| \\
&= \frac{1}{4}\left|\Phi_k^T(t)\cdot H_t^{-1}(\theta^0)\cdot\Big(h_t(\theta) - h_t(\hat{\theta}(t))\Big)\right| \\
&\leq \frac{1}{4}\left\|\Phi_k(t)\right\|_{H_t^{-1}(\theta^0)}\left\|h_t(\theta) - h_t(\hat{\theta}(t))\right\|_{H_t^{-1}(\theta^0)} \\
&\leq \frac{1}{4\,c_\sigma}\left\|\Phi_k(t)\right\|_{V_t^{-1}}\left\|h_t(\theta) - h_t(\hat{\theta}(t))\right\|_{V_t^{-1}}, \qquad (2.42)
\end{aligned}$$

where the first inequality holds by the Lipschitz property of the logistic function, which is Lipschitz with constant $1/4$. The equality holds by the mean-value theorem. In the second inequality, we use $|a^T\cdot M\cdot b| \leq \|a\|_M\,\|b\|_M$ where $\|a\|_M = \sqrt{a^T M a}$. Next, recall that $H_t(\theta^0)\succeq c_\sigma\,V_t\succ 0 \Rightarrow H_t^{-1}(\theta^0)\preceq \frac{1}{c_\sigma}\,V_t^{-1}$, which implies that the inequality $\|x\|_{H_t^{-1}(\theta^0)} \leq \frac{1}{\sqrt{c_\sigma}}\,\|x\|_{V_t^{-1}}$ holds for each vector $x\in\mathbb{R}^d$. This is used in the last inequality above.

Next, the projected estimator is obtained by $\hat{\theta}(t) = \arg\min_{\theta\in\Theta}\left\|\,h_t(\theta) - h_t(\bar{\theta}(t))\right\|_{V_t^{-1}}$. Therefore, we can obtain the following decomposition for each time period t:

$$\begin{aligned}
\left\|h_t(\theta) - h_t(\hat{\theta})\right\|_{V_t^{-1}} &= \left\|h_t(\theta) - h_t(\bar{\theta}(t)) + h_t(\bar{\theta}(t)) - h_t(\hat{\theta}(t))\right\|_{V_t^{-1}} \\
&\leq \left\|h_t(\theta) - h_t(\bar{\theta}(t))\right\|_{V_t^{-1}} + \left\|h_t(\bar{\theta}(t)) - h_t(\hat{\theta}(t))\right\|_{V_t^{-1}} \\
&\leq 2\left\|h_t(\theta) - h_t(\bar{\theta}(t))\right\|_{V_t^{-1}}.
\end{aligned}$$

Note that the first inequality holds by the triangle inequality, and the second inequality is by definition of the projected estimator $\bar{\theta}(t)$.

Plugging the above result into (2.42) completes the proof. □

Lemma II.5 (**Upper-Bound on Summation of a Subset of Feature Vectors**). Let $\{\Phi(t)\}_{t=1}^{\infty}$ be a sequence of vectors in $\mathbb{R}^{d+K}$ such that $\|\Phi(t)\|\leq 1$. Define $\Phi(0)=0$ and $V_t = \sum_{s=0}^{t-1}\Phi(s)\,\Phi^T(s)$. If there exists an integer m such that $\lambda_{min}(V_{m+1})\geq 1$, then the following bound holds almost surely for all $n>0$ (Adopted from Lemma 2 in [108]):

$$\sum_{t=m+1}^{m+n}\left\|\Phi(t)\right\|_{V_t^{-1}} \leq \sqrt{2n\,(d+K)\,\log\left(\frac{m+n}{d+K}\right)}.$$

Proof. Proof of Lemma II.5: From Lemma 11 of [1], we have the following:

$$\begin{aligned}\sum_{t=m+1}^{m+n} \|\Phi(t)\|^2_{V_t^{-1}} &\leq 2 \log\left(\frac{\det(V_{m+n+1})}{\det(V_{m+1})}\right) \\ &\leq 2\,(d+K)\,\log\left(\frac{\text{trace}(V_{m+1})+n}{d+K}\right) - 2\log\big(\det(V_{m+1})\big) \\ &\leq 2\,(d+K)\,\log\left(\frac{m+n}{d+K}\right)\end{aligned}$$

Note that $\text{trace}(V_{m+1}) = \sum_{t=1}^{m} \text{trace}\big(\Phi(t)\,\Phi^T(t)\big) = \sum_{t=1}^{m} \|\Phi(t)\|^2 \leq m$ since $\|\Phi(t)\| \leq 1$, and also we have that $\det(V_{m+1}) = \prod_{j=1}^{d+K} \lambda_j \geq \big(\lambda_{\min}(V_{m+1})\big)^{(d+K)} \geq 1$ since $\lambda_{min}(V_{m+1}) \geq 1$.

Using Cauchy-Schwartz inequality yield the following:

$$\sum_{t=m+1}^{m+n} \|\Phi(t)\|_{V_t^{-1}} \leq \sqrt{n \sum_{t=m+1}^{m+n} \|\Phi(t)\|^2_{V_t^{-1}}} \leq \sqrt{2n\,(d+K)\,\log\left(\frac{m+n}{d+K}\right)}.$$

□

Lemma II.6 (**Tail Characterization of $N(t)$**). Consider a sequence of i.i.d. non-negative random variables $\{D(t)\}_{t=1}^{T}$ with mean μ_D that satisfies the regularity condition (2.4). Define a random variable $N(t) = \sum_{s=1}^{t-1} \mathbb{1}\{s + D(s) \geq t\}$. Then, we have the following:

(a) $N(t)$ is a sub-Gaussian random variable and $N(t) \leq 2\mu_D + \tilde{\sigma}\sqrt{2\log\big(1/\delta\big)} + c$ for each time period t, with probability $1-\delta$.

(b) For the maximal quantity $N_{\max} = \max_{1\leq t\leq T} N(t)$, we have that $N_{\max} \leq 2\mu_D + \tilde{\sigma}\Big(\sqrt{2\log T} + \sqrt{2\log\big(1/\delta\big) + c'(\tilde{\sigma}\sqrt{2\log T}+1)}\Big) + c$ with probability $1-\delta$,

where $c = 2\,\tilde{\sigma}^2 \log\,(2\sigma_D^2+1) + 1$, $c' = 2\log(2\sigma_D^2+1)$ and $\tilde{\sigma} = \sigma_D\sqrt{p+2}$ (Adopted from Proposition 1 in [165]).

Lemma II.7 (**Upper-Bound on the Summation of Feature Vectors**). For the total summation of feature vectors $\Phi_k(t) \in \mathbb{R}^{d+K}$ over T time steps, the following inequality holds almost surely (Adopted from Lemma 9 in [53]):

$$\sum_{t=1}^{T} \|\Phi_k(t)\|^2_{V_t^{-1}} \leq 2\log\left(\frac{\det(V_{T+1})}{\det(\gamma I)}\right),$$

where V_t is the design matrix at time step t.

Proof. Proof of Lemma II.4: First, recall that the design matrix $V_t \in \mathbb{R}^{(d+K_1)\times(d+K_1)}$ corresponding to the first $t-1$ time-steps of the observed features is defined as:

$$V_t = \sum_{s=1}^{t-1} \Phi_k(s) \cdot \Phi_k^T(s) + \gamma I.$$

So, the determinant of $V_{T+1} = V_T + \Phi_k(T) \cdot \Phi_k^T(T)$ is obtained as follows:

$$\begin{aligned}
\det(V_{T+1}) &= \det\left(V_T + \Phi_k(T) \cdot \Phi_k^T(T)\right) \\
&= \det\left(V_T^{1/2}\left(I + V_T^{-1/2}\,\Phi_k(T) \cdot \Phi_k^T(T)\,V_{T-1}^{-1/2}\right)V_{T-1}^{1/2}\right) \\
&= \det\left(V_T\right)\,\det\left(I + \left(V_T^{-1/2}\,\Phi_k(T)\right)\left(V_T^{-1/2}\,\Phi_k(T)\right)^T\right) \\
&= \det\left(V_T\right)\left(1 + \|\Phi_k(T)\|^2_{V_T^{-1}}\right) \\
&= \det\left(\gamma I\right)\left[\prod_{t=1}^{T}\left(1 + \|\Phi_k(t)\|^2_{V_t^{-1}}\right)\right]. \qquad (2.43)
\end{aligned}$$

Note that the fourth equality holds because all the eigenvalues of a matrix of the form $(I + xx^T)$ where $x \in \mathbb{R}^n$ are one except the one eigenvalue, which is $1 + \|x\|^2$. Also, the last equality is obtained by recursion. Taking the logarithm of (2.43) from both sides results in the following:

$$\sum_{t=1}^{T} \log\left(1 + \|\Phi_k(t)\|^2_{V_t^{-1}}\right) = \log\left(\frac{\det(V_{T+1})}{\det(\gamma I)}\right).$$

Using the inequality $x \leq 2\log(1+x)$ for any $0 \leq x \leq 1$ along with the above result, we have:

$$\begin{aligned}
\sum_{t=1}^{T} \min\left\{1, \|\Phi_k(t)\|^2_{V_t^{-1}}\right\} &\leq 2\sum_{t=1}^{T} \log\left(1 + \min\left\{1, \|\Phi_k(t)\|^2_{V_t^{-1}}\right\}\right) \\
&\leq 2\sum_{i=1}^{T} \log\left(1 + \|\Phi_k(t)\|^2_{V_t^{-1}}\right) \\
&= 2\log\left(\frac{\det(V_{T+1})}{\det(\gamma I)}\right).
\end{aligned}$$

Since $\|\Phi_k(t)\|^2_{V_t^{-1}} \leq \lambda^{-1}_{\min}(V_t)\,\|\Phi_k(t)\|^2 \leq \lambda^{-1}_{\min}(V_t)$ (note we assumed $\|\Phi_k(t)\| \leq 1$) and $\lambda_{\min}(V_t) \geq 1$, then we have $\|\Phi_k(t)\|_{V_t^{-1}} \leq 1$. Accordingly, the following bound holds, which

completes the proof:

$$\sum_{t=1}^{T} \|\Phi_k(t)\|^2_{V_t^{-1}} \leq 2 \log \left(\frac{\det(V_{T+1})}{\det(\gamma I)} \right).$$

□

The following standard results and definitions in this Appendix are stated without any proof, and we refer interested readers to the chapter 2 of [152] for their detailed arguments.

Definition 3 (**Martingale**). A sequence of random variables $\{X_i\}_{i=1}^{\infty}$ is said to be a *martingale* sequence adapted to some other sequence of random variables $\{Z_i\}_{i=1}^{\infty}$ if we have the following:

1. X_i is a measurable function of $Z_1, Z_2, \cdots, Z_i$ for each i.
2. $\mathbb{E}\big[X_{i+1} | Z_1, Z_2, \cdots, Z_i\big] = X_i$, i.e., X_{i+1} is centered around X_i.
3. $\mathbb{E}\big[|X_i|\big] < \infty$ for each i.

Definition 4 (**Martingale Difference Sequence**). Assume that $\{X_i\}_{i=1}^{\infty}$ is a martingale sequence adapted to $\{Z_i\}_{i=1}^{\infty}$ and define the random variable $D_i = X_i - X_{i-1}$, then $\{D_i\}_{i=1}^{\infty}$ is called a *martingale difference sequence* adapted to $\{Z_i\}_{i=1}^{\infty}$.

Definition 5 (**Sub-Gaussian Martingale**). $\{D_i\}_{i=1}^{\infty}$ is said to be a σ^2*-sub-Gaussian martingale difference sequence* adapted to $\{Z_i\}_{i=1}^{\infty}$ if the following inequality holds almost surely:

$$\mathbb{E}\Big[\exp\left(\lambda D_i\right) | Z_1, Z_2, \cdots, Z_{i-1} \Big] \leq \exp\left(\frac{\lambda^2 \sigma^2}{2} \right), \text{ for all } \lambda \in \mathbb{R}.$$

Theorem II.8 (**Azuma-Hoeffding for Sub-Gaussian Martingale Difference Sequence**). *Assume that $\{D_i\}_{i=1}^{\infty}$ is a σ^2-sub-Gaussian martingale difference sequence adapted to $\{Z_i\}_{i=1}^{\infty}$, then the following inequalities hold:*

$$\mathbb{P}\Big(\sum_{i=1}^{n} D_i \geq t \Big) \leq \exp\left(-\frac{t^2}{2n\,\sigma^2} \right), \text{ for all } t \geq 0.$$

$$\mathbb{P}\Big(\sum_{i=1}^{n} D_i \leq -t \Big) \leq \exp\left(-\frac{t^2}{2n\,\sigma^2} \right), \text{ for all } t \geq 0.$$

2.7.4 Appendix D (Algorithm 2): UCB-based S-SGD Bandit Algorithm

In this appendix, we present the UCB-based S-SGD Bandit algorithm. Note that the theoretical performance of this algorithm is provided in Corollary II.2.

Initialization. Choose an arbitrary dosage vector $y_k(1)$ for each treatment $k \in \mathcal{K}$, where its k^{th} element belongs to $[\Delta_k^{LB}, \Delta_k^{UB}]$ and other elements are zero. Initialize the step size η_k.

Main Loop. We proceed in time periods $\mathcal{T} = \{1, \ldots, T\}$ with the following steps.

Step 1 (Context Information). Observe the context information $(\phi^{\mathcal{X}}(t), \psi^{\mathcal{X}}(t))$ of patient t.

Step 2 (Parameter Estimation). Estimate the marginal effect of actions $\hat{\pi}(t)$ according to:

$$\hat{\pi}(t) = \Big(\sum_{s \in \mathcal{M}(t)} \Upsilon_k^T(s) \cdot \Upsilon_k(s) + \eta I \Big)^{-1} \Big(\sum_{s \in \mathcal{M}(t)} \Upsilon_k^T(s) \cdot \Pi_k(s) \Big),$$

and also estimate the marginal effects of context $\bar{\theta}(t)$ according to:

$$\sum_{s \in \mathcal{M}(t)} \Big(R_k(s) - \sigma\big(\Phi_k^T(s) \cdot \bar{\theta}(t) - f_k\big(\Upsilon_k(s); \hat{\pi}(s)\big)\big) \Big) \cdot \Phi_k(s) - \kappa \bar{\theta}(t) = 0.$$

If $\bar{\theta}(t) \in \Theta$, then $\hat{\theta}(t) = \bar{\theta}(t)$, **otherwise** project $\bar{\theta}(t)$ by:

$$\hat{\theta}(t) = \arg\min_{\theta \in \Theta} \left\| h_t(\theta) - \sum_{s \in \mathcal{M}(t)} R_k(s) \cdot \Phi_k(s) \right\|_{V_t^{-1}}.$$

Step 3 (Policy Optimization and Implementation). Choose the treatment $k(t)$ along with the corresponding dosage $y_k(t)$ for patient t, where

$$k(t) = \arg\max_{k \in \mathcal{K}} \left\{ \sigma\Big(\Phi_k^T(t) \cdot \hat{\theta}(t) - f_k\big(\Upsilon_k(t);\ \hat{\pi}(t)\big)\Big) + \lambda_1(t) \left\|\Phi_k(t)\right\|_{V_t^{-1}} + \lambda_2(t) \left\|\Upsilon_k(t)\right\|_{U_t^{-1}} \right\},$$

where $\lambda_1(t) = \frac{1}{2c_\sigma}\left(\sqrt{(d_1 + K) \log\left(1 + \frac{t}{\gamma(d_1+K)}\right) + \log\left(\frac{1}{\delta^2}\right)} + \sqrt{2(d_1 + K) N(t) \log\left(\frac{N(t)}{d_1+K}\right)} + \kappa c_\theta \right)$, and also $\lambda_2(t) = \frac{\max\{\alpha_k, \beta_k\}}{2}\left(\lambda \sqrt{(d_2 + 2K) \log\left(1 + \frac{t}{\nu(d_2+2K)}\right) + \log\left(\frac{1}{\delta^2}\right)} + c_\psi \sqrt{4(d_2 + 2K) N(t) \log\left(\frac{N(t)}{d_2+2K}\right)} + \eta c_\pi \right)$.

Step 4 (Feedback Observation). For each treatment $k \in \mathcal{K}$, obtain the set $\mathcal{S}_k(t)$ as the set of time-stamps with *new realized* feedbacks (i.e., rewards and treatment-dosage sub-outcomes) at time period t, which is calculated by $\mathcal{S}_k(t) = \mathcal{M}_k(t+1) - \mathcal{M}_k(t)$, where the set $\mathcal{M}_k(t)$ contains the time-stamps with realized feedbacks by the end of time period $t-1$ corresponding to treatment k.

Step 5 (Online Stochastic Sub-Gradient Descent). Update and calculate the following:

(5a) Obtain $\widetilde{\nabla} f_k(s) = [\hat{\tau}(t)]_k \Big(\alpha_k \mathbb{1}\big\{\Pi_k(s) > q\big\} - \beta_k \mathbb{1}\big\{\Pi_k(s) < q\big\} \Big)$ for each new realized treatment-dosage sub-outcome $\Pi_k(s)$ with time-stamp $s \in \mathcal{S}_k(t)$ at period t for each treatment k.

(5b) Obtain the next period's dosage for each treatment $k \in \mathcal{K}$ by

$$[y_k(t+1)]_k = \mathbf{Proj}_{[\Delta_k^{LB}, \Delta_k^{UB}]} \Big([y_k(t)]_k - \eta_k \sum_{s \in \mathcal{S}_k(t)} \widetilde{\nabla} f_k(s) \Big).$$

2.7.5 Appendix E (Algorithm 3): TS-based B-SGD Bandit Algorithm

In this appendix, we present the TS-based B-SGD Bandit algorithm. Note that the theoretical performance of this algorithm is provided in Proposition 4.

Initialization. Initialize a safe, arbitrary dosage vector $y_k(1)$ for each treatment $k \in \mathcal{K}$, where its k^{th} element belongs to $[\Delta_k^{LB}, \Delta_k^{UB}]$ and other elements are zero. Initialize the step size η_k.

Parameters. Let m_ℓ^1 and $(q_\ell^1)^{-1}$ be the mean and variance of the Gaussian prior distribution for the ℓ-th element of θ vector. These parameters can be initialized based on some prior beliefs.

Main Loop. We proceed in time periods $\mathcal{T} = \{1, \ldots, T\}$ with the following steps.

Step 1 (Context Information). Observe the context information $(\phi^{\mathcal{X}}(t), \psi^{\mathcal{X}}(t))$ of patient t.

Step 2 (Sampling). Draw a random sample $[\tilde{\theta}(t)]_\ell$ from the posterior normal distribution of $[\theta(t)]_\ell \sim \mathcal{N}(m_\ell^t, (q_\ell^t)^{-1})$ for each corresponding element $\ell \in \{1, \ldots, d_1 + K\}$ of the feature vector.

Step 3 (Policy Optimization and Implementation). Having the sample vector $\tilde{\theta}(t)$, choose the treatment $k(t)$ with the corresponding dosage $y_k(t)$ for patient t, where

$$k(t) = \arg\max_{k \in \mathcal{K}} \left\{ \sigma\Big(\Phi_k^T(t) \cdot \tilde{\theta}(t) - f_k\big(\Psi_k(t), y_k(t)\big) \Big) \right\}.$$

Step 4 (Feedback Observation). For each treatment $k \in \mathcal{K}$, obtain $\mathcal{S}_k(t)$ as the set of time-stamps with *new realized* feedbacks at time period t, which is calculated by $\mathcal{S}_k(t) = \mathcal{M}_k(t+1) - \mathcal{M}_k(t)$, where the set $\mathcal{M}_k(t)$ contains the time-stamps with realized feedbacks by the end of time period $t-1$ corresponding to treatment k.

Step 5 (Online Bandit SGD). Update and calculate the following:

(5a) Calculate $\widetilde{g}_{k,s} = \frac{u_k(s)}{\vartheta}\ f_k\big(y_k(s)\big)$ with time-stamp $s \in \mathcal{S}_k(t)$ at time period t for each treatment $k \in \mathcal{K}$, where $u_k(s)$ is a random unit number and $0 < \vartheta < 1$.

(5b) Calculate the next period's dosage for each treatment $k \in \mathcal{K}$ by

$$[y_k(t+1)]_k = \mathbf{Proj}_{\Omega_k}\Big([y_k(t)]_k - \eta_k \sum_{s\in\mathcal{S}_k(t)} \widetilde{g}_{k,s}\Big) + \vartheta\ u_k(t),$$

where $\Omega_k = \{y \mid \frac{1}{(1-\vartheta)}y \in [\Delta_k^{LB}, \Delta_k^{UB}]\}$, $u_k(t)$ is a random unit number, and $0 < \vartheta < 1$.

Step 6 (Belief Updating). We leverage the patients' bandit feedback whose time-stamp is in $\mathcal{S}_k(t)$ for each treatment k to update the posterior distribution of θ vector parameter.

(6a) Solve the following optimization problem,

$$\hat{\rho} \triangleq \arg\max_{\rho}\ \frac{1}{2}\ \sum_{\ell=1}^{d_1+K}\ q_\ell^t\ ([\rho]_\ell - m_\ell^t)^2 + \sum_{k\in\mathcal{K}}\sum_{s\in\mathcal{S}_k(t)} \log\left(1 + e^{-R_k(s)\,(\rho^T\Phi_k(s) - f_k(y_k(s);\tilde{\pi}(t)))}\right).$$

(6b) Update the mean and variance of the posterior distribution for θ as follows:

$$m^{t+1} = \hat{\rho}, q_\ell^{t+1} = q_\ell^t + \frac{e^{-(\hat{\rho}^T\Phi_k(t) - f_k(y_k(t);\tilde{\pi}(t)))}}{(1 + e^{-(\hat{\rho}^T\Phi_k(t) - f_k(y_k(t);\tilde{\pi}(t)))})^2}\Big([\Phi_k(t)]_\ell\Big)^2,\ \forall\ \ell \in \{1,\ldots,d_1+K\}.$$

2.7.6 Appendix F: Belief Updating with Bayesian Inference

We explain the general idea of online Bayesian logistic and linear regressions. We use them in Step (6) of the proposed algorithm to adaptively update the belief about the unknown parameters.

Consider a training data set $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$, where $x_i \in \mathbb{R}^d$ is a feature vector and $y_i \in \{-1, +1\}$ (failure, success) is a response variable. Assume that the success/failure probability is a parameterized function $\mathbb{P}(y = \pm 1|x) = \sigma(y \cdot \theta^T x)$ with unknown parameter $\theta \in \mathbb{R}^d$, where the link function is chosen as the logistic function $\sigma(u) = \frac{1}{1+\exp(-u)}$. With the assumption that training labels are independently generated given θ, the likelihood is $\mathbb{P}(\mathcal{D}|\theta) = \prod_{i=1}^n \sigma(y_i \cdot \theta^T x_i)$. The estimate of θ can be found by maximizing the likelihood $\mathbb{P}(\mathcal{D}|\theta)$, or equivalently minimizing the regularized negative log-likelihood under l_2 regularization (to avoid over-fitting) with parameter $\kappa > 0$:

$$\min_{\theta}\ \ -\sum_{i=1}^{n} \log(\sigma(y_i \cdot \theta^T x_i)) + \frac{\kappa}{2}\left\|\theta\right\|.$$

It can be proved that this regularized log-likelihood function is concave in θ for logistic regression. Consequently, various optimization methods (e.g., Newton's and gradient decent algorithms) can be used for solving it. However, we have a *sequential* setting in our problem. If we want to update our estimator for a set of new realized data at each iteration, we should re-optimize the above problem using all the previous realized data, which is computationally inefficient.

To deal with this hurdle, we adopt a Bayesian approach to perform a recursive update for the estimator with each set of new realized data. Consider a prior $\mathbb{P}(\theta)$ for the parameter θ, we apply the Bayes' theorem to obtain the posterior $\mathbb{P}(\theta|\mathcal{D}) = \frac{\mathbb{P}(\mathcal{D}|\theta)\,\mathbb{P}(\theta)}{\mathbb{P}(\mathcal{D})} \propto \mathbb{P}(\mathcal{D}|\theta)\,\mathbb{P}(\theta)$. Unfortunately, exact Bayesian inference for the above linear classifier is not tractable since the evaluation of the posterior involves a product of sigmoid functions. We can either use Markov Chain Monte Carlo methods [83], or analytic approximations to the posterior [149].

We apply the Laplace approximation, which deploys a Gaussian approximation to the posterior. This can be obtained by finding the mode of the posterior distribution and then fitting a Gaussian distribution centered at that mode (see Chapter 4 of [33]). In particular, define the logarithm of the unnormalized posterior distribution:

$$\Psi(\theta|m, Q, \mathcal{D}) = \log \mathbb{P}(\mathcal{D}|\theta) + \log \mathbb{P}(\theta). \tag{2.44}$$

Since the logarithm of a Gaussian distribution is a quadratic function, we use a second-order Taylor series to Ψ in (2.44) around its MAP (maximum a posterior) solution $\hat{\theta} = \arg\max_\theta \Psi(\theta|m, Q, \mathcal{D})$:

$$\Psi(\theta) \approx \Psi(\hat{\theta}) - \frac{1}{2}(\theta - \hat{\theta})^T\, \mathbf{H}\, (\theta - \hat{\theta}), \tag{2.45}$$

where $\mathbf{H}$ is the Hessian of the negative log posterior evaluated at $\hat{\theta}$ i.e., $\mathbf{H} = -\nabla^2 \Psi(\theta)|_{\theta=\hat{\theta}}$. By exponentiating both sides of (2.45), we can observe that the Laplace approximation results in a normal approximation to the posterior i.e., $\mathbb{P}(\theta|\mathcal{D}) \approx \mathcal{N}(\theta|\hat{\theta}, \mathbf{H}^{-1})$.

For Gaussian priors $\mathbb{P}(\theta) = \mathcal{N}(\theta|m, Q)$, we have the following from (2.44):

$$\Psi(\theta|m, Q, \mathcal{D}) = -\frac{1}{2}(\theta - m)^T\, Q^{-1}\, (\theta - m) + \sum_{i=1}^{n} \log\big(\sigma(y_i \cdot \theta^T x_i)\big), \tag{2.46}$$

and the Hessian $\mathbf{H}$ evaluated at $\hat{\theta}$ is obtained as $\mathbf{H} = Q^{-1} + \sum_{j=1}^{n} \frac{e^{-\langle \hat{\theta}, x_j \rangle}}{\left(1+e^{-\langle \hat{\theta}, x_j \rangle}\right)^2}\, x_j\, x_j^T$.

Starting from a Gaussian prior $\mathcal{N}(\theta_\ell|m_\ell^1, (q_\ell^1)^{-1})$ with mean m_ℓ^1 and variance $(q_\ell^1)^{-1}$ for each $\ell \in \{1, \ldots, d\}$, the Laplace approximated posterior is $\mathcal{N}(\theta_\ell|m_\ell^t, (q_\ell^t)^{-1})$ after the t^{th}

iteration. Recently, [159] proposed an online Bayesian logistic regression algorithm that finds the MAP solution (2.44) to the posterior after observing a set of new realized data $\mathcal{S}_t = \{(x_j, y_j)\}_{j=1}^m$ at iteration t by solving the following optimization problem (via a one-dimensional bisection search method):

$$\hat{\theta} = \arg\max_{\theta} \ \frac{1}{2} \sum_{\ell=1}^{d} q_\ell^t \, ([\theta]_\ell - m_\ell^t)^2 + \sum_{j \in \mathcal{S}_t} \log\left(1 + e^{-y_j \, \langle \theta, x_j \rangle}\right).$$

The updated mean is then $m^{t+1} = \hat{\theta}$ and the inverse variance of each weight θ_ℓ is given by the curvature at the mode as $q_\ell^{t+1} = q_\ell^t + \sum_{j \in \mathcal{S}_t} \frac{e^{-\langle \hat{\theta}, x_j \rangle}}{\left(1+e^{-\langle \hat{\theta}, x_j \rangle}\right)^2} \left([x_j]_\ell\right)^2$ for each $\ell \in \{1, \ldots, d\}$ (see [159] for details).

To conduct the Bayesian inference for online linear regression, we have the same issue as described for the online logistic regression above. [6] proposed a Bayesian inference procedure to update the posterior distribution in online linear regression. In particular, consider $P^t = I_d + \sum_{i=1}^{t-1} x_i \, x_i^T$ and $u^t = (P^t)^{-1}\big(\sum_{i=1}^{t-1} x_i \, y_i\big)$, where $x_i \in \mathbb{R}^d$ is a feature vector and y_i is a response variable. Then, if the prior for parameter π at time t is given by $\mathcal{N}(u^t, (P^t)^{-1})$, then the posterior distribution for π at time $t+1$ is $\mathcal{N}(u^{t+1}, (P^{t+1})^{-1})$, which is derived as follows:

$$\begin{aligned}
\mathbb{P}(\pi | y_t) &\propto \mathbb{P}(y_t | \pi) \, \mathbb{P}(\pi) \\
&\propto \exp\Big\{ -\frac{1}{2}\big((y_t - \pi^T x_t)^2 + (\pi - u^t)^T P^t (\pi - u^t)\big)\Big\} \\
&\propto \exp\Big\{ -\frac{1}{2}\big(y_t^2 + \pi^T \, x_t \, x_t^T \, \pi + \pi^T \, P^t \, \pi - 2\pi^T \, x_t \, y_t - 2\pi^T \, P^t \, u^t\big)\Big\} \\
&\propto \exp\Big\{ -\frac{1}{2}\big(\pi^T \, P^{t+1} \, \pi - 2\pi^T \, P^{t+1} \, u^{t+1}\big)\Big\} \\
&\propto \exp\Big\{ -\frac{1}{2}\big(\pi - u^{t+1}\big)^T \, P^{t+1} \, \big(\pi - u^{t+1}\big)\Big\} \\
&\propto \mathcal{N}(u^{t+1}, (P^{t+1})^{-1}).
\end{aligned}$$

2.7.7 Appendix G: More Empirics on Selection of Third-line Medication

Tables 2.5 and 2.6 report the percentages of all different medications selected by the physicians in the ACCORD BP trial and our online learning algorithms for the success and failure data subsets, respectively.

	The trial	**TS-SGD**	**TS-AvgDos**	**TS-LowDos**	**CTS-HighDos**	**TS-TwoDim**
Reserpine	0.8%	1.2%	6.9%	3.4%	7.2%	2.53%
Chlorthalidone	38.5%	39.7%	33.8%	33.6%	33.9%	35.02%
Metoprolol	8.3%	10.3%	25.7%	28%	23.9%	14.33%
Furosemide	2.3%	2.2%	0.7%	2.4%	1.6%	3.2%
Terazosin	1.6%	1.4%	1%	4.9%	4.1%	2.4%
Carvedilol	0.5%	0.4%	1%	2.1%	3.3%	4.05%
Hydralazine	0.7%	0.4%	1%	0.7%	1.3%	0.53%
HCTZ	47.3%	44.4%	30.1%	25%	24.8%	37.95%

Table 2.5: The percentages of medications selected in the ACCORD BP trial and by the online learning algorithms for the trial's success data subset.

	The trial	**TS-SGD**	**TS-AvgDos**	**TS-LowDos**	**CTS-HighDos**	**TS-TwoDim**
Reserpine	1.2%	31.4%	30.5%	26.7%	50.1%	47.1%
Chlorthalidone	37.8%	55.2%	60.6%	58.5%	46.4%	48.8%
Metoprolol	8.7%	1.6%	1.5%	2.2%	0.3%	0.8%
Furosemide	16%	1.8%	1.1%	1.6%	0.4%	0.6%
Terazosin	0.8%	1.4%	1.5%	3.5%	0.5%	0.9%
Carvedilol	1.2%	0.7%	1.8%	2.4%	0.5%	0.5%
Hydralazine	0.9%	1%	1.6%	3%	1.1%	0.6%
HCTZ	33.4%	6.9%	1.4%	2%	0.6%	0.8%

Table 2.6: The percentages of medications selected in the ACCORD BP trial and by the online learning algorithms for the trial's failure data subset.

CHAPTER III

Online Advance Scheduling with Overtime: A Primal-Dual Approach [1]

3.1 Introduction

We study a fundamental online resource allocation problem in the context of outpatient healthcare scheduling in which a heterogeneous stream of patient or customer requests/referrals arrives one at a time with a declared reward for receiving service from one of a finite number of heterogeneous servers/providers. This is an *online adversarial* setting in which there is no knowledge about the future arrival process of customers. Our objective is to develop efficient and effective online algorithms that are *robust* to the arrival process. Upon arrival of each customer (e.g., patient, ad, job), the system chooses in real-time both a server (e.g., clinician, advertiser, machine) and a date for service over a horizon subject to capacity constraints without knowing any information of the subsequent incoming customers in the future. Customers have *heterogeneous* capacity requirements and rewards. Each server/provider has a finite regular capacity but can be expanded at the expense of *overtime* cost. For human servers, shifts and work schedules dictate the *regular* capacity. However, overtime decisions cannot be ignored in most service systems due to the uncertain arrival process and the fact that urgent and unpredictable events happen from time to time. We shall call this *Online Scheduling with Budgeted Overtime under Adversarial Arrivals* (OS-BOAA) problem.

We develop new *online algorithms* for making not only a server-date *allocation decision* for each incoming customer but also an *overtime decision* for each server on each day within a scheduling horizon. The goal is to maximize the total reward less the total overtime cost. Our proposed online algorithms are (i) both easy-to-implement and extremely efficient to

[1] Keyvanshokooh, E., Shi, C., Van Oyen, M. P. (2020), Online Advance Scheduling with Overtime: A Primal-Dual Approach. Manufacturing & Service Operations Management.

compute, and also (ii) admit a theoretical worst-case performance guarantee. To make them more valuable in applications (e.g., healthcare operations), we extend our approach to a *rolling horizon* online scheduling setting.

An algorithm is called *online* if at all points in time, exogenous future information is unknown and the algorithm has to make adaptive decisions based on current and past information. In contrast, an *offline* algorithm (hypothetically) knows all future arrival information in advance and therefore, represents a level of performance that is the theoretical best case. A natural goal is to create near-optimal policies that perform "relatively well", regardless of the actual realizations of the arrival process. We investigate our problem under the paradigm of *competitive analysis*. In the competitive analysis, no prior knowledge is given about the sequence of arrivals, nor are they assumed to follow any predictable pattern. Our online algorithms' performance is stated as a fraction of an optimal performance achieved when knowing the entire arrival stream a priori. For $r \leq 1$, if an algorithm can guarantee that this fraction is at least r for every instance of customer sequence, then it is said to achieve a *competitive ratio* of r. Our algorithms are very effective and efficient in devising *robust online policies* that can be implemented without any information about the evolution of future demands. Moreover, our empirical study shows that the proposed online policy performs much better than its theoretical worst-case performance guarantee.

3.1.1 Motivating Applications

The described online resource allocation problem is directly relevant to many fundamental applications in operations research and management science. We illustrate some applications below.

Service Scheduling. In service systems such as healthcare delivery systems, the servers may correspond to surgeons, clinicians, physicians, nurse practitioners, technicians, etc. The customers are patients whose requests for service are received over time. Upon their arrival to outpatient clinics or hospitals, patients with different urgency levels must be able to obtain a clinic or surgery appointment while servers should also be kept available for future possible higher urgency patients. The servers are often *extensible* in the sense that they can be utilized for longer than the normal available time, but with a cost of *overtime*. Typically, they have a planned utilization time (e.g., a capacity of 8 hours per day), beyond which overtime begins to accrue. If no capacity is available within the planned working hours of servers for serving the request, overtime is used to serve patients and alleviate the excess workload of healthcare delivery system. Hence, the addition of an overtime decision is important to appointment scheduling as another layer of operational decisions.

Online Advertising. In online advertising, the servers corresponding to advertisers, and the capacity of each server is associated with the advertiser's budget. When a user browses a web-page, an impression request is sent to the ad platform (e.g., Google, Yahoo, Microsoft). Ad impressions arrive sequentially and randomly over time. Each impression, depending on its features, requests a known non-negative bid (value) from each advertiser. Upon arrival of an impression request, the ad platform must allocate it immediately and irrevocably to an advertiser for displaying an ad. The advertising platform then earns the bid, and the budget of the advertiser is often depleted by the same amount. The goal is to maximize the total reward (revenue) of all allocations subject to a budget constraint without knowing future impression requests. Advertisers choose a limited daily budget based on their advertising goals. When the budget is limited, their ads might not be shown as frequently as they would like. However, considering an additional budget that is analogous to our model's *overtime* can help get their ads displayed on highly profitable days that have high viewing traffic, and are therefore expensive days to advertise.

Make-To-Order (MTO) Flexible Manufacturing. With servers corresponding to manufacturing cells (or machines, plants), customers request varying and possibly unique products over time. Manufacturing may start only after a request is received. A flexibility structure governs which cells can produce which types of requests. Clearly, the naive strategy of always accepting a product request as long as the capacity is available is myopic and sub-optimal, because higher value product may be turned away when capacity is exhausted. The firm must decide whether to accept or reject a request (without knowing the future requests) and, if accepted, allocate it to a feasible cell on a day within the horizon. Particularly when orders are tied to long-term customers with urgent jobs, the firm may be better off utilizing *overtime* capacity in order to avoid delaying or rejecting jobs and therefore losing customers and market share.

Our models and algorithms can be tailored to other applications such as revenue management (e.g., room/seat allocation in hotels/airlines), the airline industry (e.g., the online arrival process of customers who purchase tickets), online auctions (e.g., eBay in which the arrival of bidding customers for an auction can be properly modeled using our online framework), online retailers (e.g., the real-time orders that Amazon receives and must be filled quickly), and call-center operations.

In the above-mentioned applications, uncertainty about future demand poses a major challenge that makes the resource allocation problem very complex. Most traditional resource allocation models assume that the underlying probability distribution of customer arrivals is fully or at least partially known as a *priori* knowledge. There exist many approaches for solving such optimization problem under uncertainty. Stochastic optimization, where

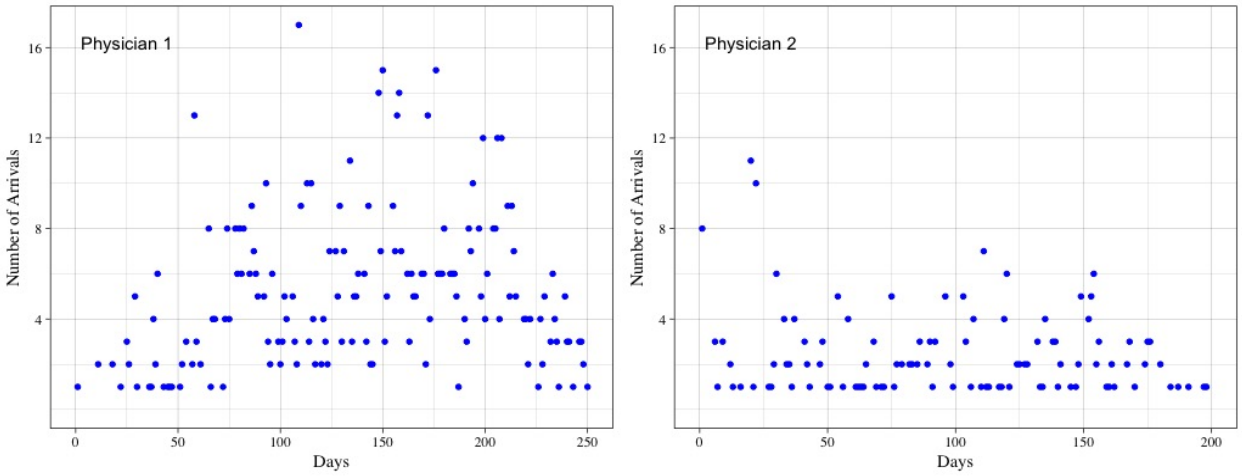


Figure 3.1: The number of patient request arrivals to two physicians over different days in a clinic of our partner health system. The arrival process is non-stationary and does not follow any clear pattern (see case study in §3.6).

probabilistic distributions characterize uncertain parameters, is an approach that has been widely applied. However, in many practical settings, it is unrealistic to expect that the distribution of customer arrivals can be precisely characterized. Misspecification of this distribution could lead to meaningless results as well.

In many service systems, the demand is highly unpredictable or cannot be learned a *priori* due to unforeseeable components such as traffic spikes and a competitor's change of strategy. In healthcare systems, we may set up a new appointment scheduling system for a clinic/hospital and lack sufficient data to precisely estimate the arrival distribution. Figure 3.1 shows high variability and uncertainty in the number of patient arrivals to two physicians over time in a medical clinic of our partner health system (see the case study in §3.6). In online advertising, an unpredictable breaking news story could flood the ad platform with impression requests from news websites. The stream of impression requests is then *non-stationary* and does not follow any clear pattern. Hence, more conservative approaches are needed. A popular conservative approach is robust optimization, where the uncertainty is characterized by an uncertainty set, rather than distribution. Although arguably more robust than stochastic optimization, it again depends heavily on the structure of the uncertainty set. In this paper, we make no assumption about the arrival distribution and exploit *online optimization* as the most robust approach to optimization under uncertainty, which is more appropriate for sequential decision making problems where *a priori* assumption about the structure of the problem data uncertainty is not available and/or not reliable.

3.1.2 Main Results and Contributions

Our main results and contributions are summarized as follows.

We propose an online algorithm (Algorithm 1) for the OS-BOAA problem. We shall call it *Heterogeneous Online Optimization Procedure with Budgeted OverTime* (HOOP-BOT). Our proposed online policy is (i) robust to future information, (ii) easy-to-implement and extremely efficient to compute, (iii) allows for heterogeneity in both rewards and service requirements, and (iv) admits a theoretical competitive ratio (CR) (see Definition 6). Theorem III.1 gives the CR of the proposed algorithm. In particular, the ratio of service requirement per request to total capacity $R_{max} \to 0$, the CR becomes $r = 1/(1+\theta^*)$ where we find *a closed-form expression* for the coefficient θ^* (see Theorem III.1 for details). When the reward is proportional to the service requirement for each customer, the worst-case CR is $r \geq \frac{e-1}{e+\sqrt{e}} \simeq 0.4$, regardless of input parameters. In addition, as the maximum overtime cost $d_{\max}$ decreases to 0, r improves to $(1-\frac{1}{e}) \simeq 0.633$ at the rate of $O(1/d_{\max})$. Theorem III.2 states that no algorithm can achieve a better CR than $(1-\frac{1}{e}) \simeq 0.633$.

For practical implementation purposes, we also extend our online algorithms to a *rolling horizon setting* (Algorithm 2), which enables practitioners to make allocation-overtime decisions for every incoming customer and each server on each day in a rolling horizon fashion. We shall call it *Rolling Heterogeneous Online Optimization Procedure with Budgeted OverTime* (R-HOOP-BOT).

Our OS-BOAA problem is closely related to the online Adwords problem with concave rewards, and the proposed online algorithms are based on online primal-dual methods. In this regard, the closest studies to ours are [39], [58], and [47]. In the following, we summarize our high-level approaches and techniques, and also discuss the major departures of this paper from prior literature.

(1) Incorporating overtime decisions. Besides making the usual allocation decision for each customer, our model makes an *overtime* or *capacity expansion* decision for each server on each day over a scheduling horizon. In many applications such as healthcare operations, overtime cannot be ignored due to uncertain arrivals and the fact that urgent and unpredictable events happen from time to time. Thus, it is important to leave some flexibility for these events by means of capacity expansion (at the expense of incurring some overtime cost). The current online scheduling algorithms have not incorporated overtime decisions in their decision-making (see [158, 148, 70, 157], and references therein).

Closely related to our model with overtime decisions, [58] and [47] generalize the online primal-dual method of [39] for the online Adwords problem with concave rewards where the objective function is assumed to be *monotonically non-decreasing concave*. They derive a parametric CR of $r \geq F$ where a numeric value for F is calculated by solving a set of differ-

ential equations for each problem instance. The non-decreasing monotonicity assumption is crucial for their technical CR analysis because through this assumption, they ensure that (i) dual solution obtained by their online algorithm is feasible in each iteration, and (ii) the set of differential equations has a solution for each instance, and so a CR is derived. However, this monotonicity assumption is clearly *violated* in our setting with overtime. Hence, we cannot replicate their algorithm and competitive analysis, which spurs new methodological innovations.

(2) Multiple multiplicative rules for obtaining dual prices of capacity constraints. Our algorithms use online primal-dual frameworks of [41]. The introduction of the overtime decision (or capacity expansion) makes the competitive ratio analysis of the HOOP-BOT (Algorithm 1) invariably harder compared to the existing online algorithms in the literature.

The main departure from prior works is that the dual typically has a single price, while our online primal-dual method has two different dual prices due to the addition of *overtime*. So, the existing results of [41] and [39] do not extend into our setting with overtime. Consequently, to resolve this difficulty and solve the online linear program (LP), we develop *three multiplicative updating rules* for computing dual prices of capacity constraints including (i) one for the *non-overtime* case, (ii) one for the transition from the *non-overtime to overtime* case, and (iii) one for the *overtime* case (see Figure 3.3 for these three cases). This new idea requires a different proof strategy for deriving the CR of the proposed online primal-dual paradigm (see Lemmas III.3 and III.4, as well as Propositions 6, 7, 8, and 9). In other words, there are two parameters θ_1 and θ_2 corresponding to two multiplicative updating rules such that one parameter for the non-overtime case tilts the CR up, and the other one for the overtime case tilts the CR down. Thus, we introduce a *balancing point*, which is the *min-max* value between these two parameters (see Figure 3.4 for calculating the balancing point), and derive *closed-form* expressions for these parameters. Moreover, the HOOP-BOT (Algorithm 1) obtains a *feasible dual* solution but an *almost feasible primal* solution. To compute the competitive ratio, we need to construct a feasible primal solution from this almost feasible primal solution, and next bound the ratio of their objective values (see Proposition 9).

(3) Applicability to real-world healthcare operations practices. Since this research was conducted in collaboration with a partner health system, a particular emphasis has been put on real-world applicability of our proposed methods. We test the validity of our methods and measure their impact using actual data (see §3.6). To this aim, we evaluate the empirical performance of our online policies by comparing to two common benchmarks: (i) First-Come-First-Served (FCFS) policy and (ii) Nested Threshold Policy with Overtime (NTPO). We develop NTPO, which is a more sensible class of policies than FCFS policy and

based on the idea of protecting urgent patients with some pre-specified threshold levels to reserve capacity a priori and serving them with overtime (if necessary). Our empirical results illustrate that the proposed online policies work consistently well over a range of parameter choices, and they outperform the FCFS and NTPO policies by a large margin in terms of both the empirical CR and the number of days a patient is deferred.

For practical implementation purposes, our model is generalized in two ways. First, we extend our online algorithms to a *rolling horizon paradigm*. Most online algorithms either accept or reject a customer upon his/her arrival. But, in most healthcare settings, rejection would not be a suitable outcome of serving a patient. To deal with this issue, we cast a rejection as a *deferred decision*. Specifically, suppose we cannot or choose not to accommodate an incoming patient within a scheduling horizon with respect to her/his arrival day. Then, we defer making an allocation decision until the next day in the R-HOOP-BOT (Algorithm 2). We investigate the importance of having a rolling horizon framework on real data (see §3.6.3 of the case study).

Second, to model more practical scenarios, our model allows customers to have *heterogeneity* not only in service requirements but also in reward/urgency values which might be different than the service requirements. This is in contrast to the current literature surveyed below (e.g., [39] have *homogeneous* reward values and service requirements), making the competitive analysis of online algorithms challenging (see §3.4). In healthcare, reward/urgency values can be a vector indicating how the reward/urgency depends on the provider with whom the appointment is made. This feature allows us to set the reward to zero for patients who should not be seen by a particular physician while being set to higher values for the most appropriate physicians. This can help create an appointment scheduling system where physicians spend more time *"practicing at the top of their license"* as we allocate each patient to the most appropriate physician [143]. This relatively new term captures the idea that it is ideal for physicians to service the most skilled interventions for which they are trained and licensed.

3.1.3 Related Literature

Our work is related to three research domains and streams of literature, namely, online Adwords problem, online primal-dual approach, and appointment scheduling problem.

Online Adwords Problem. Our models are related to works on the online Adwords or bipartite matching problem. In this problem, one set of nodes is known and corresponds to the set of finite servers. Demand requests arise one at a time and correspond to the second set of nodes. When a demand node arrives, its adjacency to the server nodes is revealed with its edge weights. The arriving demand needs to be matched irrevocably to an available

server (if any). The objective is to maximize the total value of the matching. There is no overtime assumption that is modeled. This problem has been studied under different online input models. Customer arrival models broadly belong to two categories: *stochastic models* and *adversarial models*.

There is a stream of studies about the stochastic models. The main studies include [100], [75], [18], [65], [94], and [117] with random permutation assumption, [66], [19], and [90] with known i.i.d arrival distribution, and [59] with unknown i.i.d arrival distribution. Stochastic online algorithms either (i) depend heavily on a *precise forecast* for the arrival patterns of incoming customers using a vast amount of historical data, or (ii) assume that arrival pattern can be *perfectly learned* as we observe the sequence of incoming customers. Hence, these online algorithms are not able to react quickly to sudden changes in the arrival pattern. This is particularly the case in a new market or when a known system or market is in a period of upheaval. Instead, the *adversarial model* has the key advantage of being *robust* to any change in the arrival process because it does not exploit any forecast or learning tool. Thus, the distribution of arriving demands can fluctuate over time.

Developing online algorithms for the adversarial models is typically more challenging due to having no prior assumption on the arrival pattern (see [95] and [4]). Under the adversarial setting, our paper is closely related to the online Adwords problem introduced by [126]. They develop a 0.633-competitive algorithm based on a trade-off revealing LP technique. [39] then propose a classical primal-dual paradigm for this problem and derive the same CR. [58] and [47] generalize the work of [39] in which the objective function is assumed to be monotonically non-decreasing concave. The key differences between these works and ours were discussed in §3.1.2.

Online Primal-Dual Approach. The main theoretical tool employed is *the primal-dual framework*. This method is a powerful algorithmic scheme that has proved to be extremely applicable in a wide range of problems in the area of approximation algorithms (see [161]). These applications include inventory control problem [107], single-machine scheduling problem [81] and capacitated facility location problem [42].

The primal-dual scheme has been extended to the setting of *online* algorithms ([41]). The basic idea is that the LP and its dual program guide the decisions made by the online algorithm. Specifically, it simultaneously constructs online both (feasible or almost feasible) primal and dual solutions. This online scheme has applications in make-to-order production systems, routing, machine load balancing, and packing-covering problems (see [40] and [41]). Our paper is closely related to [39], which design online primal-dual algorithms with CR of 0.633 for the online Adwords problem. Our online primal-dual method departs from this study (see discussions in §3.1.2).

Appointment Scheduling Problem. Appointment scheduling is a ubiquitous operation in service systems, especially in healthcare operations. Comprehensive surveys are provided by [79], and [7]. Recently, much progress has been made (see e.g., [46], [118], [119], [162], [109], and [112]). Here, we present the ones that we believe are the most relevant to our work.

A part of the literature considers *multi-day* scheduling, in which patients are dynamically assigned to appointment days (e.g., [110]). According to [116], *allocation scheduling* and *advance scheduling* are the two main paradigms for multi-day scheduling. In allocation scheduling, the number of patients to be served today is determined, and the rest of the patients are added to a waitlist (see e.g., [157], [88], [17], and [72]). Among these works, [157] is the only study that develops a 2-competitive online algorithm, for scheduling arriving patients in the context of allocation scheduling with cancellation. Recently, more works have focused on advance scheduling paradigm, in which patients are scheduled into future days at the time of their arrival (see e.g., [150], [73], [137], [80], and [64]). Our work is an advance scheduling model as well.

Perhaps the closest scheduling works to ours are [158], [148], and [70]. [158] develop a 0.5-competitive online algorithm for advance scheduling, and their model is useful when the patient preferences are revealed before determining appointment day. [148] propose a 0.321-competitive online algorithm for the advance reservation of service with the goal of maximizing the total expected capacity utilization. [70] develop a 0.5-competitive online algorithm for choosing an assortment of services to display to arriving customers. However, our paper significantly departs from these online algorithms (see §3.1.2 for a detailed discussion on critical differences between these papers and ours).

3.1.4 Organization

The rest of this paper is organized as follows. In §3.2, we present our model, discuss the offline optimization problem, and define the performance measure. In §3.3, we present our main algorithm and results. In §3.4, we present the competitive analysis. In §3.5, we extend our online algorithms to a rolling horizon paradigm. In §3.6, we implement our online algorithms on appointment-scheduling data obtained from a medical clinic of our partner health system. In §3.7, we conclude the paper and presents some avenues for future research.

3.2 Online Scheduling Problem with Overtime

We describe the general problem statement for the OS-BOAA problem, present the offline optimization problem, and introduce the performance measure (competitive ratio).

3.2.1 The Problem Statement

We study an online resource allocation problem with *budgeted overtime* in an *adversarial* setting in the sense that there is no prior knowledge about the distribution of the number of customer arrivals on a given time span such as a day. Furthermore, we consider an *online* setting, which means that the service requests are handled as they arrive, and allocation decisions are provided in an online fashion without information on future requests. Unlike most prior studies (see e.g., [72]), we do not require the scheduler to wait until the end of each day and then make *batch*-type allocation decisions for all arrivals on that day. Instead, as soon as a customer request is received, the scheduler makes an instantaneous and irrevocable allocation decision for this customer. We use the terms patient and customer interchangeably.

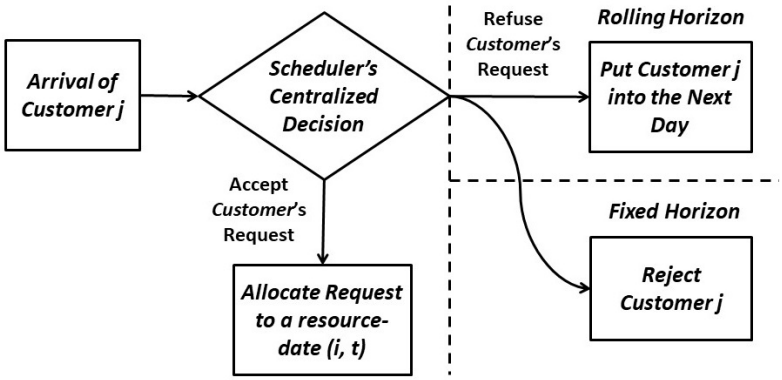


Figure 3.2: Decision process for making real-time server-date allocation decision by a centralized scheduler in the online scheduling problem with budgeted overtime under adversarial arrivals.

On the first day of a T-period horizon, when customer requests arrive sequentially to the system (e.g., a medical clinic), the system either accepts or rejects each customer as they arrive. If we accept a customer, we should make a scheduling decision immediately; that is, we assign this customer to (i) a specific server/provider and (ii) a service date within the T-period scheduling horizon. Customers request service but do not ask for specific dates explicitly. In §3.5, we extend our proposed online algorithms to have a *rolling horizon approach* in which the rejection decision turns into a decision to *defer* service, meaning that we either cannot or choose not to accommodate this incoming customer within the scheduling horizon with respect to her/his arrival day. Instead, we delay making the scheduling decision for this customer until the next day, and the option to defer is again allowed on each subsequent day. Figure 3.2 illustrates the decision process for making online server-date allocation decision for each customer with the option to defer this decision.

We use $[n] = \{1, \ldots, n\}$ and $[m] = \{1, \ldots, m\}$ to denote the set of n servers/providers

and the set of m customers/patients, respectively. Note that n is known *a priori*, but m is not. The scheduling horizon has T days, indexed by $t \in [T] = \{1, 2, \ldots, T\}$. Customers arrive sequentially on the first day of the horizon, i.e., $t = 1$, and we schedule their service on any day $t \in [T]$. For each day $t \in [T]$, each server $i \in [n]$ has a limited *regular* capacity $B_{i,t} \in \mathbb{R}_+$ (e.g., a clinician may be required to work for 8 hours per day as her/his regular schedule), and also a limited *overtime* budget $U_{i,t} \in \mathbb{R}_+$ (e.g., a clinician may work maximum 4 hours overtime if an emergency arises). For each server $i \in [n]$, let $d_i \in \mathbb{R}_+$ denote its (per-unit-of-time) overtime cost. Note that servers correspond to clinicians (physicians/physician assistants) in our case study in §3.6.

For each arriving customer $j \in [m]$, we need information that includes his/her (i) service time and (ii) (urgency-based) reward. The service time and the reward of each customer $j \in [m]$ could be *heterogeneous* for each server $i \in [n]$. Stochasticity in service time and reward is not considered in our analysis, but we incorporate that into our sensitivity analysis in §3.6.2. Let $b_{i,j} \in \mathbb{R}_+$ denote the service time of customer j if the customer is assigned to server i and $c_{i,j} \in \mathbb{R}_+$ denote the (urgency-based) reward of customer j if this customer is assigned to server i. Their dependency is permitted for generality, allowing us to discourage or encourage certain pairings. Both vectors $\mathbf{b}_j = (b_{1,j}, \ldots, b_{n,j})$ and $\mathbf{c}_j = (c_{1,j}, \ldots, c_{n,j})$ are not known *a priori*, and are only realized upon arrival of each customer j. Indeed, when cast as a optimization problem, one can see that the constraints are not given up front, and they are revealed one by one corresponding to each customer. However, we add an almost innocuous assumption that there exists a $\gamma_{\min} > 0$ (and the scheduler knows the value), which is the minimum ratio between the reward and service requirement. More precisely, we assume $c_{i,j}/b_{i,j} \geq \gamma_{\min}$ for each $i \in [n]$ and $j \in [m]$. One can think of this as analogous to knowing a lower bound on the minimum wage. This assumption allows us to assume that, without loss of generality, $c_{i,j}/b_{i,j} \geq 1$ for each $i \in [n]$ and $j \in [m]$, since we can uniformly scale the units of $c_{i,j}$ and d_i by multiplying a constant $1/\gamma_{\min}$ for all i and j. This unit transformation will not affect the CR.

Upon arrival of customer j, the scheduler *immediately* chooses both a provider and a day over the scheduling horizon without knowing the subsequent arrivals. That is, we need to make a binary decision $y_{i,j,t} \in \{0, 1\}$ for allocating an incoming customer j to a server $i \in [n]$ on day $t \in [T]$. If server $i \in [n]$ on day $t \in [T]$ is allocated to customer j, exactly $b_{i,j}$ units of server i must be provided, and the reward earned is $c_{i,j}$. In case that the regular capacity runs out, the scheduler may choose to schedule overtime as long as it does not exceed the remaining overtime capacity. Let $v_{i,t}$ be the overtime committed for server $i \in [n]$ on day $t \in [T]$. Our online algorithm makes both an *allocation decision* for each incoming customer and an *overtime decision* for each server on each day. The objective is to maximize the

cumulative reward less the cumulative overtime cost.

To ease our presentation, we introduce some *key notation* as follows. We let

$$\begin{aligned}
&\gamma_{\max} = \max_{i,j} \{c_{i,j}/b_{i,j}\}, \quad \gamma_{\min} = \min_{i,j} \{c_{i,j}/b_{i,j}\}, \\
&d_{\max} = \max_{i}\{d_i\}, \quad d_{\min} = \min_{i}\{d_i\}, \\
&R_{\max} = \max_{i,j,t}\{b_{i,j}/B_{i,t}\}, \quad \alpha = (1+R_{\max})^{1/R_{\max}}.
\end{aligned} \tag{3.1}$$

Note that $\gamma_{\max}$ is the maximum ratio of reward to service requirement among all i and j, and $R_{\max}$ is the maximum ratio of service requirement per request to its total capacity among all i and j and t. We will repeatedly use these quantities throughout the paper.

3.2.2 Offline Optimization Model with Overtime

We first formulate an offline optimization model (P) where the full information of all incoming customers/patients for the OS-BOAA problem is given *a priori*.

Primal Problem ($\boldsymbol{P}$):

$$\max_{y,v} \quad \sum_{t=1}^{T}\sum_{i=1}^{n}\sum_{j=1}^{m} c_{i,j}y_{i,j,t} - \sum_{t=1}^{T}\sum_{i=1}^{n} d_i v_{i,t} \tag{3.2}$$

$$\text{s.t.} \quad \sum_{t=1}^{T}\sum_{i=1}^{n} y_{i,j,t} \leq 1, \quad \forall j \in [m] \tag{3.3}$$

$$\sum_{j=1}^{m} b_{i,j}y_{i,j,t} \leq B_{i,t} + v_{i,t}, \quad \forall i \in [n], t \in [T] \tag{3.4}$$

$$v_{i,t} \leq U_{i,t}, \quad \forall i \in [n], t \in [T] \tag{3.5}$$

$$v_{i,t} \geq 0, \quad y_{i,j,t} \in \{0,1\}, \quad \forall i \in [n], t \in [T], j \in [m]. \tag{3.6}$$

Constraint (3.3) ensures that each customer is allocated to one server-date and constraint (3.4) limits the capacity of each server on each day. Constraint (3.5) limits the overtime capacity of each server on each day. If we relax the binary constraints to continuous ones in (P), we obtain a natural LP relaxation of (P). We take the dual of this LP relaxation, by defining the dual variables z_j, $x_{i,t}$ and $u_{i,t}$ corresponding to constraints (3.3), (3.4) and (3.5), respectively:

Dual Problem (D):

$$\min_{x,u,z} \sum_{t=1}^{T}\sum_{i=1}^{n}(B_{i,t}x_{i,t} + U_{i,t}u_{i,t}) + \sum_{j=1}^{m} z_j \tag{3.7}$$

$$\text{s.t. } b_{i,j}x_{i,t} + z_j \geq c_{i,j}, \quad \forall i \in [n], t \in [T], j \in [m] \tag{3.8}$$

$$x_{i,t} - u_{i,t} \leq d_i, \quad \forall i \in [n], t \in [T], j \in [m] \tag{3.9}$$

$$x_{i,t}, u_{i,t}, z_j \geq 0, \ \forall i \in [n], t \in [T], j \in [m]. \tag{3.10}$$

Next, we develop an online algorithm based on a primal-dual approach which admits a performance guarantee in order to solve the online version of the model (P) and its corresponding dual model (D). We use the offline model (P) as a *benchmark* to evaluate the performance of the proposed online primal-dual algorithms.

3.2.3 Performance Measure

Competitive analysis is the most widely used method for evaluating the performance of online algorithms [34]. It considers the relative performance between an online algorithm and an optimal offline algorithm under the worst-case input instance. An *online algorithm* addresses a sequential decision-making problem, where individual model inputs are received in an online fashion, and an immediate decision on each successive input must be made at the time the input is received without taking future inputs into account. Once a decision is made regarding an individual input, it cannot be revoked. On the other hand, an *offline algorithm* treats the entire input sequence as given in advance and then optimizes.

Under the assumption of a maximization problem, let Ω_{ALG} denote the set of all possible input sequences of customer arrivals to an online algorithm ALG, and let $\omega = (j_1, j_2, \cdots) \in \Omega_{ALG}$ be a sample path of customer arrivals. An offline algorithm knows the whole arrival input $\omega = (j_1, j_2, \cdots)$ at time zero, but at any time t an online algorithm only knows the realization of all the arrivals prior to t. Let $Z_{ALG}(\omega)$ denote the objective value achieved by the online algorithm ALG for the input $\omega \in \Omega_{ALG}$, and $Z_{OPT}(\omega)$ be the objective value of an optimal offline algorithm.

Definition 6 (**Competitive Ratio**).

- *If for any problem instance* $\omega = (j_1, j_2, \cdots) \in \Omega_{ALG}$, $Z_{ALG}(\omega) \geq r \, . \, Z_{OPT}(\omega)$ *where* $0 \leq r \leq 1$, *then the online algorithm ALG is called* r*-competitive.*

- *The (asymptotic) competitive ratio of the online algorithm ALG is then defined as:*

$$r = \inf_{\omega \in \Omega_{ALG}} \left\{ \frac{Z_{ALG}(\omega)}{Z_{OPT}(\omega)} \right\}.$$

- *Larger competitive ratios imply better online algorithms for maximization problem.*

Thus, the competitive ratio of an online algorithm is a guarantee on a certain level of performance.

3.3 Online Algorithm for Online Scheduling with Overtime

We consider the *online* version of (P) in which the information of each customer (including service time and reward parameters) is only revealed sequentially upon arrival. We develop an online primal-dual algorithm to solve the online version of (P). The customers sequentially arrive to the system on the first day $t = 1$ and an irrevocable allocation decision is made upon each arrival.

Algorithm 1 Online Primal-Dual Algorithm for Online Scheduling with Overtime (HOOP-BOT)

Initially set the dual prices $x_{i,t}, v_{i,t} = 0, u_{i,t} = 1$; $\forall i \in [n],\ t \in [T]$.
for each arriving customer j **do**
  Assign customer j to server i^* on day/slot t^* such that $(i^*, t^*) = \arg\max_{i \in [n], t \in [T]} \{ c_{i,j} - b_{i,j} \cdot x_{i,t} \}$.
  if $(c_{i^*,j} - b_{i^*,j} \cdot x_{i^*,t^*}) \geq 0$ **then**
    Set $z_j \leftarrow c_{i^*,j} - b_{i^*,j} \cdot x_{i^*,t^*}$ **and** $y_{i^*,j,t^*} \leftarrow 1$;
    if $0 \leq x_{i^*,t^*} < d_{i^*}$ **then**
      Set $x_{i^*,t^*} \leftarrow x_{i^*,t^*} \left(1 + \frac{b_{i^*,j}}{B_{i^*,t^*}}\right) + \theta_1 \left(\frac{c_{i^*,j}}{B_{i^*,t^*}}\right)$ [*Updating Rule* (I)].
      if $x_{i^*,t^*} > d_{i^*}$ **then set** $x_{i^*,t^*} \leftarrow d_i$ [*Updating Rule* (II)].
    else:
      Set $u_{i^*,t^*} \leftarrow u_{i^*,t^*} \left(1 + \frac{b_{i^*,j}}{B_{i^*,t^*} + U_{i^*,t^*}}\right) + \theta_2 \left(\frac{\bar{c}_{i^*,j}}{B_{i^*,t^*} + U_{i^*,t^*}}\right)$ [*Updating Rule* (III)].
      Set $x_{i^*,t^*} \leftarrow u_{i^*,t^*} + d_{i^*}$ **and** $v_{i^*,t^*} \leftarrow v_{i^*,t^*} + b_{i^*,j}$.
  else:
    Set $y_{i,j,t}, \leftarrow 0,\ z_j \leftarrow 0;\ \forall i \in [n],\ t \in [T]$.

3.3.1 Description of the Online Primal-Dual Algorithm 1 (HOOP-BOT)

We provide a high-level description of the HOOP-BOT (online primal-dual Algorithm 1). The dual prices $x_{i,t}$ and $u_{i,t}$ are initialized at zero and one, respectively. Upon arrival of a customer j, an optimization problem (*acceptance/rejection criterion*) in Step 3 is solved to figure out which server-date (i^*, t^*) to allocate this customer. If we accept serving the customer, we either use the regular capacity (Steps 6-8) or the overtime of server-date (i^*, t^*) (Steps 9-11). If we assign customer j to *regular* capacity, we update the dual price x_{i^*,t^*} of regular capacity. So, its value is gradually updated by the *incrementally increasing updating rule* (I), and if the updated quantity x_{i^*,t^*} becomes greater than d_{i^*}, we set it to d_{i^*} by using the *updating rule* (II). However, if we use *overtime* to serve customer j, then we update the dual price u_{i^*,t^*} of overtime by the *incrementally increasing updating rule* (III) and update the dual price x_{i^*,t^*} by adding d_{i^*} to the updated price u_{i^*,t^*}. Figure 3.4 shows a graphical presentation of how the dual price x_{i^*,t^*} is gradually constructed by using the proposed multiplicative updating rules (I), (II) and (III) as customers sequentially arrive into the system. In the online Algorithm 1, the coefficient $\bar{c}_{i,j}$ is defined as $\bar{c}_{i,j} = c_{i,j} - d_i\, b_{i,j}$. Note that θ_1 and θ_2 in multiplicative updating rules (I) and (II) are two coefficients that need to be determined.

3.3.2 Main Results of the Online Primal-Dual Algorithm 1 (HOOP-BOT)

We present our main theoretical result of the HOOP-BOT for solving the online scheduling problem with budgeted overtime under adversarial arrivals and discuss it below.

Theorem III.1 (**Competitive Ratio Analysis of the HOOP-BOT**). *With the parameters α, $R_{\max}$, $d_{\max}$, and $\gamma_{\max}$ defined in (3.1), the online primal-dual Algorithm 1 (HOOP-BOT) admits the following competitive ratio*

$$r \;=\; \frac{1 - \;2\gamma_{\max} R_{\max}}{1+\theta^*},$$

where

$$\theta^* \;=\; \frac{(1+R_{\max})(\zeta+\lambda) \;+\; \sqrt{(1+R_{\max})^2(\zeta+\lambda)^2 + 4(1+R_{\max})(\alpha-(1+R_{\max}))\zeta\lambda}}{2(\alpha-(1+R_{\max}))}, \tag{3.11}$$

$\lambda = \max\big(\max_{i,j}\big\{\frac{c_{i,j}}{b_{i,j}} - d_i\big\}, 0\big)$, *and* $\zeta = \min\left(d_{\max}, \gamma_{\max}\right)$.

In particular, when $R_{\max} \to 0$*, the competitive ratio simplifies to* $r = 1/(1+\theta^{**})$ *where*

$$\theta^{**} = \frac{(\zeta+\lambda) + \sqrt{(\zeta+\lambda)^2 + 4(e-1)\zeta\lambda}}{2(e-1)}. \tag{3.12}$$

It is worthwhile pausing to discuss the interpretations and implications of Theorem III.1.

Proportional Case and Discussion of Results. While our results hold for the general case, for ease of presentation, we interpret our theoretical results using a special case where the reward is proportional to the amount of service requirement for each customer, i.e., $c_{i,j} = k\ b_{i,j}$ for all i, j for some fixed integer constant $k > 0$. This special case is a rough simplification and holds in settings in which the reward rate per usage duration is the same across different servers (e.g., all service providers work at the same hourly rate). In this case, we can readily transform the optimization problem and the online primal-dual Algorithm 1 into equivalent ones with re-definitions of $c_{i,j} \leftarrow b_{i,j}$ and $d_i \leftarrow d_i/k$. By this transformation, we have $\gamma_{\max} = 1$, $\lambda = \max(1 - d_{\min}, 0)$, and $\zeta = \min(d_{\max}, 1)$.

We can interpret the resulting competitive ratio r and gain practical intuition as follows. The competitive ratio r depends on the scale factor $R_{\max}$, which is the largest ratio of a service requirement per request to the total regular capacity on any day in the scheduling horizon. $R_{\max}$ is a scale factor that reflects the size of a service system relative to a single service, and the value of $R_{\max}$ is determined by how large the scale of a service operation is. The system approaches a fluid model in a sense that service requirements become infinitesimal relative to capacity, as $R_{\max}$ goes to zero. This can apply if the capacity refers to an entire service department or medical clinic.

On the one hand, when the scale of a service operation is *large*, then $R_{\max}$ is typically small. For example, if a centralized system schedules 10 identical providers with each having 8 working hours, and each service takes 1 hour on average, then $R_{\max} = 1/80$, which is very small. Our theoretical result asserts that as $R_{\max} \to 0$, the worst-case competitive ratio is

$$r = \frac{1}{1+\theta^{**}} \geq \frac{e-1}{e+\sqrt{e}} \simeq 0.4.$$

Moreover, as $d_{\max} \to 0$, the competitive ratio r improves to $(1 - \frac{1}{e}) \simeq 0.633$ at the rate of $O(1/d_{\max})$. Note that even for the special case of the model without overtime, no online algorithms can achieve a competitive ratio better than $(1 - \frac{1}{e}) \simeq 0.633$ (see [92]).

On the other hand, when the scale of a service operation is *small*, we still have meaningful results. For example, if specialist clinic schedules only one provider having 8 working hours, and each visit takes 1 hour on average, then $R_{\max} = 1/8$, which is not small. When $R_{\max} = 1/8$, the theoretical worst-case competitive ratio is $r \geq 0.253$. However, we show that the

empirical performance is much better in our numerical study conducted in §3.6 (where we use the the value of $R_{\max} = 1/8$ and the empirical r is high as shown in Tables 3.2 and 3.5).

Methodologically, the online primal-dual Algorithm 1 develops new ideas as follows.

1. The introduction of the overtime (or capacity expansion) decision makes the competitive analysis of HOOP-BOT (online Algorithm 1) challenging, and as discussed in §3.1.2, the analyses used in [58], [39] and [47] cannot be extended to our online setting. To resolve this difficulty, we develop *three multiplicative updating rules* for calculating the dual prices of the capacity constraints: (i) the *non-overtime* case, (ii) the transition from the *non-overtime to overtime* case, and (iii) the *overtime* case (see Figure 3.3 for our proof strategy using different updating rules). This is unlike typical applications of the primal-dual approach, which have a single updating rule for constructing the dual prices of capacities. Having multiple updating rules requires introducing a *"switching price"* (see Definition 7) to figure out when a server/provider should switch from the use of regular capacity to overtime. It also helps to bound the ratio of changes in dual and primal values, so this ratio does not blow up (see Proposition 6).

2. Another important consideration in our proof strategy is that we need to determine two coefficients θ_1 and θ_2 corresponding to multiplicative updating rules (I) and (III), respectively. The CR depends on both coefficients. The key observation is that the coefficient θ_1 for the *non-overtime case* tilts the CR up, and the coefficient θ_2 for the *overtime case* tilts the CR down. This motivates introducing a *"balancing point"* θ^* which is the min-max value $\theta^* = \min(\max(\theta_1, \theta_2))$ between these two coefficients (see Lemmas III.3 and III.4, and Proposition 8). This idea results in a new proof strategy for deriving the CR, and we (i) show that a *unique* coefficient can be obtained for θ^* and the CR depends only on θ^*; and (ii) calculate a *closed-form* expression for computing the coefficient θ^* (see Figure 3.4). Note that when $\theta = \theta_1$ is used in the CR, it means that there is infinite capacity (i.e., $U_{i,t} = \infty$) on the amount of overtime each server may have on each day.

3. The HOOP-BOT (online Algorithm 1) obtains a *feasible dual* solution but an *almost feasible primal* solution at each iteration. To compute the CR, we need to construct a feasible primal solution from this almost feasible primal solution and bound the ratio of their objective values (see Proposition 9). This step in the proof of deriving the CR is crucial because the HOOP-BOT accepts the last customer while there is not enough regular capacity or overtime for serving this customer.

4. Our theoretical result is also independent of the size of the overtime budget. The

implication is that if the overtime is strictly not allowed in some stringent scenarios, our algorithms can handle this case. Also, if the overtime is abundant, the optimal offline policy presumably does much better than our online policy, but we are not losing much. Unlike previous studies in which they have homogeneous rewards and capacity requirement, we remove this highly restrictive limitation. Allowing *heterogeneity* in both rewards and capacity requirements of servers enables us to model much more practical online resource allocation problems.

In addition to our main result (Theorem III.1) on the lower bound for the CR of the HOOP-BOT, we also prove an upper bound for the CR of any online algorithm that can be developed for the OS-BOAA problem. In Theorem III.2, by suitably adapting the worst-case examples of [95] and [126] to our setting, we show that an upper bound of $(1-1/e) \simeq 0.633$.

Theorem III.2 (**Upper Bound for the CR**). *The competitive ratio of any randomized online algorithms is at most* $(1-1/e) \simeq 0.633$ *for the OS-BOAA problem.*

The proof of Theorem III.2 is provided in Appendix B. Admittedly, there exists a small gap between the developed lower bound for the competitive ratio of our online algorithm and the upper bound for the competitive ratio of any online algorithm. We tend to believe that our online algorithm could potentially admit a better lower bound. In the current competitive analysis, we compute a balancing point θ^* involving θ_1 and θ_2, which works well given the current logic flow. However, this balancing point is not necessarily optimal (in terms of jointly optimizing over θ_1 and θ_2), which could potentially result in a small loss in the competitive ratio. There could be a better proof strategy of determining an optimal θ^* that further smooths the transition from the region using regular capacity (parameterized by θ_1) and the region of using overtime (parameterized by θ_2). Also, there might be a tighter upper bound as a result of some "better" bad instance. Both questions spur new methodological development, and we shall leave them for future research.

3.4 Competitive Analysis of Algorithm 1 (HOOP-BOT)

We analyze the HOOP-BOT (online Algorithm 1) for solving the online version of the model (P) with budgeted overtime under adversarial arrivals. The proposed online Algorithm 1 is based on a primal-dual paradigm which maintains a set of dual variables that guides the primal solution. The evolution of the primal solution in turn determines how the dual variables are progressively updated. Let $f_P^{(j)}$ and $g_D^{(j)}$ be the primal and dual objective function values at iteration j, and $(y_{i,j,t}^{(j)}, v_{i,t}^{(j)})$ and $(x_{i,t}^{(j)}, u_{i,t}^{(j)}, z_j^{(j)})$ denote the primal and dual solutions for each server-date (i,t) at iteration j, respectively. As a road-map, the

competitive analysis of the online primal-dual Algorithm 1 is divided into the following four main parts:

(1) In each iteration j, the ratio $\Delta g_D^{(j)}/\Delta f_P^{(j)}$ of changes in the dual and primal objective function values is upper-bounded by either $(1+\theta_1)$ or $(1+\theta_2)$ (see Definition 7 and Proposition 6).

(2) The dual solution obtained by online Algorithm 1 is always *feasible* (see Proposition 7).

(3) The primal solution obtained by online Algorithm 1 is *almost feasible* (see Lemmas III.3 and III.4, and Proposition 8).

(4) We derive the competitive ratio of online Algorithm 1 (see Proposition 9).

Before proceeding with the competitive analysis of online Algorithm 1, we first define a *"switching price"* for each server-date (i,t) upon arrival of a patient j as follows.

Definition 7 (**Switching Price**). *For patient j, $P_j(i,t) = \left(\frac{B_{i,t}\, d_i - \theta_1\, c_{i,j}}{B_{i,t}+b_{i,j}}\right)$ is called a switching price for dual price $x_{i,t}$ of regular capacity for server-date (i,t) after which the overtime of server-date (i,t) is used to serve patients if patient j is allocated to server-date (i,t).*

According to online Algorithm 1, the dual price $x_{i,t}$ of regular capacity of server-date (i,t) is increasing as patients are allocated to server-date (i,t). The switching price for each server-date (i,t) then determines when the server should start using overtime to serve patients.

Proposition 6 (Bound on Dual-Primal Change Ratio). At each iteration j of the online Algorithm 1, the ratio $\Delta g_D^{(j)}/\Delta f_P^{(j)}$ of changes in the dual and primal objective function values is upper-bounded by either $(1+\theta_1)$ or $(1+\theta_2)$ depending on the value of switching price and whether regular capacity or overtime is used to serve patient j, respectively.

Proof. Proof of Proposition 6: Assume that patient j is allocated to server-date (i,t) by the *acceptance criterion* in Step 4 of online Algorithm 1. We now bound the ratio of $\Delta g_D^{(j)}/\Delta f_P^{(j)}$ at each iteration j where either $\Delta f_P^{(j)} = c_{i,j}$ if regular capacity is used to serve patient j, or $\Delta f_P^{(j)} = \bar{c}_{i,j} = c_{i,j} - d_i b_{i,j}$ if overtime is used to serve patient j, and $\Delta g_D^{(j)} = B_{i,t}\Delta x_{i,t}^{(j)} + U_{i,t}\Delta u_{i,t}^{(j)} + z_j^{(j)}$ for different cases of updating dual prices $x_{i,t}$, $u_{i,t}$ and z_j. As introduced by Definition 7, we have a *critical value* of $x_{i,t}$ or *switching price* $P_j(i,t)$ for any patient j such that if we update the dual price $x_{i,t}$ by the multiplicative updating rule (I), we achieve a

new $x_{i,t}$ value which is greater than $d_i + 1$. If we use the multiplicative updating rule (I) at iteration j for updating $x_{i,t}$ and have the following:

$$x_{i,t}^{(j)} = x_{i,t}^{(j-1)} \left(1 + \frac{b_{i,j}}{B_{i,t}}\right) + \theta_1 \left(\frac{c_{i,j}}{B_{i,t}}\right) > d_i,$$

then $x_{i,t}^{(j-1)} > P_j(i,t) = \left(\frac{B_{i,t}\ d_i - \theta_1\ c_{i,j}}{B_{i,t}+b_{i,j}}\right)$. We use this switching price $P_j(i,t)$ to investigate the ratio $\Delta g_D^{(j)}/\Delta f_P^{(j)}$ in the following three cases depending on which multiplicative updating rule is used:

Case 1: Assume $0 \le x_{i,t}^{(j-1)} < P_j(i,t)$ at iteration j; we then increase $x_{i,t}$ multiplicatively by using the updating rule (I) as $x_{i,t}^{(j)} = x_{i,t}^{(j-1)}(1 + \frac{b_{i,j}}{B_{i,t}}) + \theta_1(\frac{c_{i,j}}{B_{i,t}})$, and have $\Delta f_P^{(j)} = c_{i,j}$ and the following:

$$\Delta g_D^{(j)} = B_{i,t}\left(\left(\frac{b_{i,j}}{B_{i,t}}\right)x_{i,t}^{(j-1)} + \theta_1\left(\frac{c_{i,j}}{B_{i,t}}\right)\right) + (c_{i,j} - b_{i,j}x_{i,t}^{(j-1)}) \le (1+\theta_1)\ c_{i,j}.$$

Thus, $\Delta g_D^{(j)} \le \Delta f_P^{(j)}(1+\theta_1)$ for iteration j.

Case 2: Assume $P_j(i,t) \le x_{i,t}^{(j-1)} < d_i$ at iteration j, we then set the dual price $x_{i,t}^{(j)} = d_i$ by using the updating rule (II), and have $\Delta f_P^{(j)} = c_{i,j}$ and the following:

$$\begin{aligned} \Rightarrow \quad \Delta g_D^{(j)} &= B_{i,t}\left(d_i - x_{i,t}^{(j-1)}\right) + c_{i,j} - b_{i,j}x_{i,t}^{(j-1)} \\ \Rightarrow \quad \Delta g_D^{(j)} &\le B_{i,t}d_i + c_{i,j} - (B_{i,t}+b_{i,j})\left(\frac{B_{i,t}\ d_i\ -\ \theta_1\ c_{i,j}}{B_{i,t}+b_{i,j}}\right) \le (1+\theta_1)\ c_{i,j}. \end{aligned}$$

Thus, $\Delta g_D^{(j)} \le \Delta f_P^{(j)}(1+\theta_1)$ for iteration j.

Case 3: Assume $x_{i,t}^{(j-1)} \ge d_i$ at iteration j, we then start increasing multiplicatively the overtime dual price $u_{i,t}^{(j)}$ by using the multiplicative updating rule (III) and also increasing the dual price $x_{i,t}^{(j)}$ by $x_{i,t}^{(j)} = u_{i,t}^{(j)} + d_i$ such that:

$$\Delta u_{i,t}^{(j)} = u_{i,t}^{(j)} - u_{i,t}^{(j-1)}, \ \ \Delta x_{i,t}^{(j)} = x_{i,t}^{(j)} - x_{i,t}^{(j-1)} = (u_{i,t}^{(j)} + d_i) - (u_{i,t}^{(j-1)} + d_i) = u_{i,t}^{(j)} - u_{i,t}^{(j-1)}.$$

Hence, we have $\Delta u_{i,t}^{(j)} = \Delta x_{i,t}^{(j)}$ in this case. Then, we have $\Delta f_P^{(j)} = \bar{c}_{i,j}$ and the following:

$$\Delta g_D^{(j)} = (B_{i,t}+U_{i,t})\left(\frac{b_{i,j}u_{i,t}^{(j-1)}}{B_{i,t}+U_{i,t}} + \frac{\theta_2\ \bar{c}_{i,j}}{B_{i,t}+U_{i,t}}\right) + \left(c_{i,j} - b_{i,j}(u_{i,t}^{(j-1)} + d_i)\right) \le (1+\theta_2)\bar{c}_{i,j}.$$

Thus, $\Delta g_D^{(j)} \leq \Delta f_P^{(j)}(1+\theta_2)$ for iteration j.

Therefore, the dual-primal change ratio $\Delta g_D^{(j)}/\Delta f_P^{(j)}$ is upper-bounded by either $(1+\theta_1)$ or $(1+\theta_2)$ depending on whether regular capacity or overtime is used to serve patient j, respectively. □

Proposition 7 (Dual Feasibility Guarantee). The online primal-dual Algorithm 1 obtains a dual feasible solution $(x_{i,t}^{(j)}, u_{i,t}^{(j)}, z_j^{(j)})$ for each server-date (i,t) at each iteration j.

Proof. Proof of Proposition 7: At each iteration j, the online Algorithm 1 sets the dual price $z_j = c_{i,j} - b_{i,j}x_{i,j}$ which is ≥ 0 due to the *acceptance criterion* at Step 4. So, constraint (3.8) is satisfied. Also, constraint (3.9) is always satisfied because when the regular capacity is used, the online Algorithm 1 makes sure that $x_{i,t} \leq d_i$ and $u_{i,t} = 1$, and when the overtime capacity is used, the online Algorithm 1 makes sure that $x_{i,t} = u_{i,t} + d_i$. □

We next prove that the online primal-dual Algorithm 1 provides an *almost feasible* primal solution $(y_{i,j,t}^{(j)}, v_{i,t}^{(j)})$ at each iteration j. It is clear that the constraint (3.3) is always satisfied. To prove the almost feasibility of the primal solution, assume that the dual price $x_{i,t}$ is updated at iterations $j = 1, 2, \ldots, K_0, K_1, K_1+1, \ldots, K_2$ where updates in iterations $j = 1, 2, \ldots, K_0$ are done by using the multiplicative updating rule (I) for *non-overtime case*, update in iteration $j = K_1$ is done by using the updating rule (II) for *transition from non-overtime case to overtime case*, and updates in iterations $j = K_1+1, \ldots, K_2$ are done by using the multiplicative updating rule (III) for *overtime case*. Figure 3.3 guides our proof strategy for proving almost feasibility of primal solution, and shows how the multiplicative updating rules (I), (II) and (III) are used to update the dual price $x_{i,t}$ initialized at zero. Note that $x_{i,t}^{(K_0)}$ is the last value of $x_{i,t}$ obtained by the multiplicative updating rule (I). If we used the multiplicative updating rule (I) again on this value $x_{i,t}^{(K_0)}$, we would get the value

$$\tilde{x}_{i,t}^{(K_1)} = x_{i,t}^{(K_0)}\left(1 + \frac{b_{i,j}}{B_{i,t}}\right) + \theta_1\left(\frac{c_{i,j}}{B_{i,t}}\right),$$

which may exceed d_i. Thus, we set $x_{i,t}^{(K_1)} = d_i$ by the updating rule (II). Note that $x_{i,t}^{(K_2)}$ is the last value of $x_{i,t}$ obtained by the multiplicative updating rule (III).

Lemma III.3 (Primal Almost-Feasibility for Regular Capacity Usage Only). Given that the regular capacity $B_{i,t}$ of server-date (i,t) is used up on iteration K_1, the coefficient θ_1 in the multiplicative updating rule (I) needs to set $\theta_1 = \frac{\zeta}{(\alpha-1)}$, where $\zeta = \min(d_{\max}, \gamma_{\max})$ such that

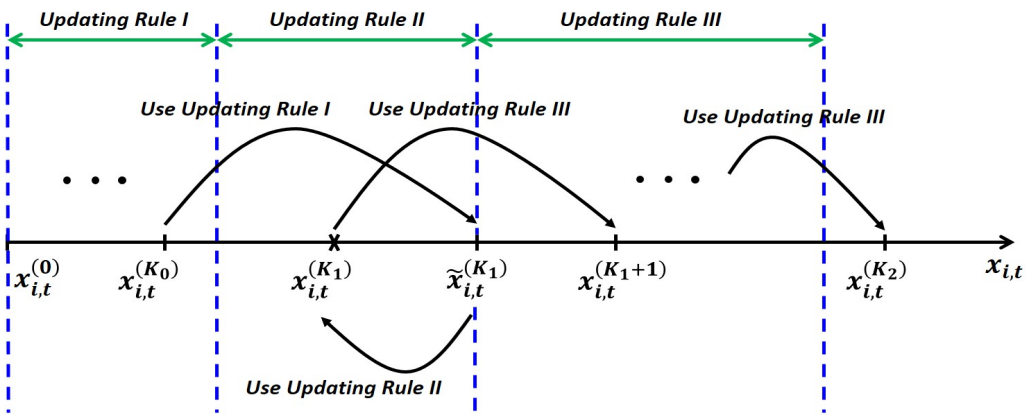


Figure 3.3: Proof strategy for primal almost feasibility: updating multiplicative schemes (I), (II) and (III) for constructing the dual price $\boldsymbol{x_{i,t}}$ for the capacity of server-date $(\boldsymbol{i},\boldsymbol{t})$ over different iterations.

the following statements hold simultaneously depending on overtime cost d_i and $\gamma_{\max}$:

$$\text{If} \quad \sum_{j=1}^{K_1} b_{i,j}\, y_{i,j,t} \;>\; B_{i,t}, \quad \text{then} \quad \tilde{x}_{i,t}^{(K_1)} \;>\; d_i, \;\text{ for each } i \in [n] \text{ where } d_i \leq \gamma_{\max}. \tag{3.13}$$

$$\text{If} \quad \sum_{j=1}^{K_1} b_{i,j}\, y_{i,j,t} \;>\; B_{i,t}, \quad \text{then} \quad c_{i,K_1} - b_{i,K_1}\, \tilde{x}_{i,t}^{(K_1)} \;<\; 0, \;\text{ for each } i \in [n] \text{ where } d_i > \gamma_{\max}. \tag{3.14}$$

Proof. Proof of Lemma III.3: From the multiplicative updating rule (I) for updating the dual price $x_{i,t}$ and $c_{i,j} \geq b_{i,j}$, we have the following for the value of $x_{i,t}^{(j)}$ at any iteration $j = 1, ..., K_0$:

$$x_{i,t}^{(j)} = x_{i,t}^{(j-1)} \left(1 + \frac{b_{i,j}}{B_{i,t}}\right) + \theta_1 \left(\frac{c_{i,j}}{B_{i,t}}\right) \quad \Rightarrow \quad \left(x_{i,t}^{(j)} + \theta_1\right) \geq \left(x_{i,t}^{(j-1)} + \theta_1\right) \alpha^{\left(\frac{b_{i,j}}{B_{i,t}}\right)},$$

where we use the technical inequality $\frac{1}{m}\ln(1+m) \geq \frac{1}{n}\ln(1+n)$ for any $0 \leq m \leq n \leq 1$. Using the above updating formula recursively until we reach the initial value $x_{i,t}^{(0)} = 0$, we have:

$$\left(\tilde{x}_{i,t}^{(K_1)} + \theta_1\right) \geq \left(\underbrace{x_{i,t}^{(0)}}_{0} + \theta_1\right) \alpha^{\sum_{j=1}^{K_1}\left(\frac{b_{i,j}}{B_{i,t}}\right)} \quad \Rightarrow \quad \tilde{x}_{i,t}^{(K_1)} > (\alpha - 1)\,\theta_1. \tag{3.15}$$

The last inequality is due to the definition of K_1 such that $\sum_{j=1}^{K_1}\left(\frac{b_{i,j}}{B_{i,t}}\right) y_{i,j,t} > 1$ and $y_{i,j,t} = 1$

for $j = 1, 2 \ldots, K_1$. Now, for (3.13) and (3.14) to hold, by (3.15), it suffices to have

$$(\alpha - 1)\ \theta_1 \ \geq\ d_i \text{ for each } i \in [n] \text{ where } d_i \leq \gamma_{\max};$$
$$(\alpha - 1)\ \theta_1 \ \geq\ \frac{c_{i,j}}{b_{i,j}} \text{ for each } i \in [n] \text{ where } d_i > \gamma_{\max},$$

where the first condition implies that we accept to serve a patient by using overtime, and the second condition implies that we reject a patient.

So, an appropriate choice is to set $\theta_1 = \frac{\zeta}{(\alpha-1)}$ where $\zeta = \min(d_{\max}, \gamma_{\max})$. □

Remark. By Lemma III.3, the coefficient θ_1 is determined such that when the regular capacity of server-date (i,t) is used up, depending on overtime cost d_i and $\gamma_{\max}$, either no more patients are allocated to server-date (i,t), or additional patients have to be served by overtime from now on if they are allocated to server-date (i,t). Next, Lemma III.4 implies that when both regular and overtime capacities are used up, no further patients can be served.

Lemma III.4 (Primal Almost-Feasibility for Regular plus Overtime Capacity Usage). Suppose that the capacity of server-date (i,t) is expanded at overtime cost. When both regular and overtime capacities $(B_{i,t}, U_{i,t})$ are used up on iteration K_2, then both coefficients in multiplicative updating rules (I) and (III) are obtained by the "balancing point" $\theta^* = \min_{\theta_1}(\max(\theta_1, \theta_2))$ defined in (3.22) such that not only statements (3.13) and (3.14) hold, but also the following statement holds:

$$\text{If } \sum_{j=1}^{K_2} b_{i,j}\ y_{i,j,t}\ >\ B_{i,t} + U_{i,t}, \quad \text{then } \ c_{i,K_2} - b_{i,K_2}\ x_{i,t}^{(K_2)}\ <\ 0, \text{ for each } i \in [n]. \tag{3.16}$$

In particular, the balancing point θ^* is obtained by the closed-form expression (3.11).

Proof. Proof of Lemma III.4: We need to find appropriate θ_1 and θ_2 such that all statements (3.13), (3.14) and (3.16) hold simultaneously. From the multiplicative updating rule (III) for updating the dual price $u_{i,t}$, we have the following relation for any iteration $j = K_1 +$

$1, \cdots, K_2$:

$$
\begin{aligned}
u_{i,t}^{(j)} &= u_{i,t}^{(j-1)} \left(1 + \frac{b_{i,j}}{B_{i,t}+U_{i,t}}\right) + \theta_2 \left(\frac{\bar{c}_{i,j}}{B_{i,t}+U_{i,t}}\right) \\
\Rightarrow\ u_{i,t}^{(j)} &\geq u_{i,t}^{(j-1)} \left(1 + \frac{b_{i,j}}{B_{i,t}+U_{i,t}}\right) + \beta_i\, \theta_2 \left(\frac{b_{i,j}}{B_{i,t}+U_{i,t}}\right) \\
\Rightarrow\ \left(u_{i,t}^{(j)} + \beta_i\theta_2\right) &\geq \left(u_{i,t}^{(j-1)} + \beta_i\theta_2\right) \left(1 + \frac{b_{i,j}}{B_{i,t}+U_{i,t}}\right) \\
\Rightarrow\ \left(u_{i,t}^{(j)} + \beta_i\theta_2\right) &\geq \left(u_{i,t}^{(j-1)} + \beta_i\theta_2\right) \bar{\alpha}^{\left(\frac{b_{i,j}}{B_{i,t}+U_{i,t}}\right)},
\end{aligned}
$$

where $\beta_i = \min_j \left\{\frac{\bar{c}_{i,j}}{b_{i,j}}\right\}$, $\bar{\alpha} = (1+\bar{R}_{max})^{1/\bar{R}_{max}}$ in which $\bar{R}_{max} = \max_{i,j,t}\{b_{i,j}/(B_{i,t}+U_{i,t})\}$. If we use the above recursion for iterations K_1+1 until K_2, then we have the following (note $\bar{\alpha} \geq \alpha$ because $h(x) = (1+x)^{(1/x)}$ is a decreasing function in x):

$$
\Rightarrow\ \left(u_{i,t}^{(K_2)} + \beta_i\theta_2\right) \geq \left(u_{i,t}^{(K_1)} + \beta_i\theta_2\right) \alpha^{\sum_{j=K_1+1}^{K_2} \left(\frac{b_{i,j}}{B_{i,t}+U_{i,t}}\right)}.
$$

From Step 11, we set the dual price $x_{i,t} = u_{i,t} + d_i$, and therefore we have the corresponding recursion for the dual price $x_{i,t}$:

$$
\left(x_{i,t}^{(K_2)} - d_i + \beta_i\theta_2\right) \geq \left(x_{i,t}^{(K_1)} - d_i + \beta_i\theta_2\right) \alpha^{\sum_{j=K_1+1}^{K_2} \left(\frac{b_{i,j}}{B_{i,t}+U_{i,t}}\right)} \tag{3.17}
$$

where $x_{i,t}^{(K_1)} = d_i$. Next, we multiply (3.15) obtained in Lemma III.3 and (3.17) together, which gives

$$
\left(\tilde{x}_{i,t}^{(K_1)} + \theta_1\right) \left(x_{i,t}^{(K_2)} - d_i + \beta_i\theta_2\right) \geq \beta_i\, \theta_1\, \theta_2\, \alpha^{\sum_{j=1}^{K_2} \left(\frac{b_{i,j}}{B_{i,t}+U_{i,t}}\right)}. \tag{3.18}
$$

Then, due to the definition of K_2 such that $\sum_{j=1}^{K_2} \left(\frac{b_{i,j}}{B_{i,t}+U_{i,t}}\right) y_{i,j,t} > 1$ and $y_{i,j,t} = 1$ for $j = 1, 2 \ldots, K_2$, we can rewrite (3.18) as:

$$
\left(\tilde{x}_{i,t}^{(K_1)} + \theta_1\right) \left(x_{i,t}^{(K_2)} - d_i + \beta_i\theta_2\right) \geq \beta_i\, \theta_1\, \theta_2\, \alpha. \tag{3.19}
$$

Next, we find an upper bound for dual price $\tilde{x}_{i,t}^{(K_1)}$. Note that $\tilde{x}_{i,t}^{(K_1)}$ is obtained if $x_{i,t}^{(K_0)}$ is updated by the updating rule (I) and $P_{K_0}(i,t) \leq x_{i,t}^{(K_0)} \leq d_i$. Applying the multiplicative updating rule (I) on the dual price $x_{i,t}^{(K_0)}$ at iteration K_1, we obtain an upper bound of $d_i \left(1 + \frac{b_{i,K_1}}{B_{i,t}}\right) + \theta_1 \left(\frac{c_{i,K_1}}{B_{i,t}}\right)$ for $\tilde{x}_{i,t}^{(K_1)}$. Replacing the dual price $\tilde{x}_{i,t}^{(K_1)}$ with this upper bound

in (3.19), we obtain the following:

$$x_{i,t}^{(K_2)} \geq \frac{\beta_i\,\theta_1\,\theta_2\,\alpha}{\left(1+\frac{b_{i,K_1}}{B_{i,t}}\right)d_i+\left(1+\frac{c_{i,K_1}}{B_{i,t}}\right)\theta_1} - \beta_i\,\theta_2 + d_i. \tag{3.20}$$

In order to have $c_{i,K_2} - b_{i,K_2}\,x_{i,t}^{(K_2)} < 0$, or $x_{i,t}^{(K_2)} > \frac{c_{i,K_2}}{b_{i,K_2}}$, we require to have the following:

$$\begin{aligned}
\Rightarrow \quad & \frac{\beta_i\,\theta_1\,\theta_2\,\alpha}{\left(1+\frac{b_{i,K_1}}{B_{i,t}}\right)d_i+\left(1+\frac{c_{i,K_1}}{B_{i,t}}\right)\theta_1} - \beta_i\,\theta_2 + d_i > \left(\frac{c_{i,K_2}}{b_{i,K_2}}\right) \\
\Rightarrow \quad & \theta_2 \geq \frac{\left(\max_j\left\{\frac{c_{i,j}}{b_{i,j}}\right\}-d_i\right)}{\beta_i}\left(\frac{\left(1+\frac{b_{i,K_1}}{B_{i,t}}\right)\,(\theta_1+d_i)}{(\alpha-1)\,\theta_1-d_i-\left(\frac{b_{i,K_1}}{B_{i,t}}\right)(\theta_1+d_i)}\right) \\
\Rightarrow \quad & \theta_2 \geq \max_{i\in[n]}\left\{\frac{\left(\max_j\left\{\frac{c_{i,j}}{b_{i,j}}\right\}-d_i\right)}{\beta_i}\left(\frac{\left(1+\frac{b_{i,K_1}}{B_{i,t}}\right)\,(\theta_1+d_i)}{(\alpha-1)\,\theta_1-\;d_i-\left(\frac{b_{i,K_1}}{B_{i,t}}\right)(\theta_1+d_i)}\right)\right\} \\
\Rightarrow \quad & \theta_2 = \lambda\left(\frac{(1+R_{\max})\,\theta_1+(1+R_{\max})\,\zeta}{(\alpha-(1+R_{\max}))\,\theta_1\;-(1+R_{\max})\,\zeta}\right),
\end{aligned}$$

where $\zeta = \min(d_{\max}, \gamma_{\max})$ and $\lambda = \max\left(\max_{i,j}\left\{\frac{c_{i,j}}{b_{i,j}} - d_i\right\}, 0\right)$. Note $\beta_i = \min_j\left\{\frac{c_{i,j}}{b_{i,j}} - d_i\right\} \geq 1$ whenever the overtime is used for serving a patient. That is because when the overtime is required, the *acceptance criterion* becomes $(c_{i,j} - b_{i,j}\,(u_{i,t} + d_i)) \geq 0$, and also $u_{i,t} \geq 1$, then we have $(c_{i,j} - d_i b_{i,j} - b_{i,j}) \geq 0$ or $\beta_i \geq 1$. So, we replace β_i by its lower bound in the formula of θ_2. Clearly, the coefficient θ_2 is a function of the coefficient θ_1, i.e., $\theta_2 = g(\theta_1)$ as shown by the above equation.

Figure 3.4 shows $\theta_2 = g(\theta_1)$ as a function of θ_1. First note that if we plug $\theta_1 = \frac{1}{(\alpha-1)}\,\zeta$ obtained by Lemma III.3 in θ_2, it yields a negative θ_2. Thus, we choose $\max(\theta_1, \theta_2)$, corresponding to the bold lines in Figure 3.4, to make sure that all statements (3.13), (3.14) and (3.16) hold simultaneously. So, all θ values on $\max(\theta_1, \theta_2)$ (i.e. the bold line in Figure 3.4) present the set of plausible θ values to choose from. Among all such plausible values, we want the smallest coefficient by $\theta^* = \min_{\theta_1}(\max(\theta_1, \theta_2))$, which leads to a *tighter* CR. This is because if we decrease θ from θ^* over the bold line, this tilts the CR up, and if we increase θ from θ^* over the bold line, this tilts the CR down (noting that the CR is inversely proportional to $(1+\theta)$, see Proposition 9). Thus, we introduce a *balancing point* θ^*, which is the min-max value between these two cases, and provides a robustness. More precisely, we find θ^* which is the intersection between $g(\theta_1)$ and θ_1 as shown in Figure 3.4. We set

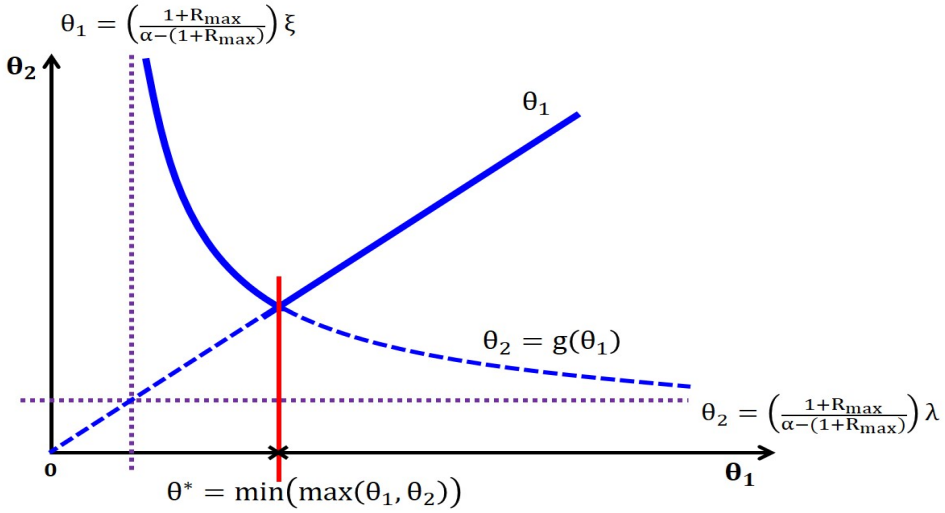


Figure 3.4: Illustration of the balancing point θ^*: finding the unique coefficient $\boldsymbol{\theta^* = \min(\max(\theta_1, \theta_2))}$ as a function of $\boldsymbol{\theta_1}$ and $\boldsymbol{\theta_2}$ coefficients. The bold line presents the set of plausible θ values to choose from. If we decrease θ from θ^* over the bold line, this tilts the CR up, and if we increase θ from θ^* over the bold line, this tilts the CR down. So, θ^* is the balancing or max-min point between these two extremes.

$\theta_2 = g(\theta_1) = \theta_1$, and solve for θ_1 to find the *balancing point* θ^*:

$$(\alpha - (1 + R_{\max}))\, \theta_1^2 \; - \; (1 + R_{\max})(\zeta + \lambda)\, \theta_1 \; - (1 + R_{\max})\, \zeta\, \lambda \; = \; 0. \tag{3.21}$$

$$\Rightarrow \quad \theta^* \; = \; \frac{(1 + R_{\max})(\zeta + \lambda) \; + \; \sqrt{(1 + R_{\max})^2(\zeta + \lambda)^2 + 4(1 + R_{\max})(\alpha - (1 + R_{\max}))\, \zeta\, \lambda}}{2(\alpha - (1 + R_{\max}))}. \tag{3.22}$$

So, if the coefficient θ^* is chosen for θ_2 in updating rule (III), then the statement (3.16) holds. □

Proposition 8 (Primal Almost-Feasibility Guarantee). The online Algorithm 1 obtains an almost primal feasible solution $(y_{i,j,t}, v_{i,t})$ for each server-date (i, t) and customer j.

Proof. Proof of Proposition 8: The proof of Proposition 8 directly follows Lemmas III.3 and III.4. □

We are now ready to derive the competitive ratio of the HOOP-BOT (online primal-dual Algorithm 1). To this aim, we need to first construct a feasible primal solution from the almost-feasible solution of the primal and then bound the ratio of this primal objective function value and the optimal objective function value of an oracle that knows the entire arrival stream.

Proposition 9 (Competitive Ratio). Let $\tilde{f}_P$ be the primal objective value of the feasible solution obtained by converting the almost feasible solution of online Algorithm 1, and f^* be the primal objective value of an optimal offline policy, then the CR of the HOOP-BOT (Algorithm 1) is:

$$r = \frac{\tilde{f}_P}{f^*} \geq \frac{1 - 2\gamma_{\max} R_{\max}}{1 + \theta^*}. \tag{3.23}$$

In particular, when $R_{\max} \to 0$, the coefficient θ^* is simplified to θ^{**} in (3.12), and $r = \frac{1}{1+\theta^{**}}$.

Proof. Proof of Proposition 9: We have proved in Proposition 6 that either $\Delta g_D^{(j)} / \Delta f_P^{(j)} \leq (1 + \theta_1)$ in the case of using the multiplicative updating rules (I) and (II), or $\Delta g_D^{(j)} / \Delta f_P^{(j)} \leq (1 + \theta_2)$ in the case of using the multiplicative updating rule (III) happens in each iteration j. Since $g_D = \sum_{j=1}^{K_2} \Delta g_D^{(j)}$ and $f_P = \sum_{j=1}^{K_2} \Delta f_P^{(j)}$, we have the following in general:

$$\frac{g_D}{f_P} \leq (1 + \theta^*) \quad \text{where } \theta^* \text{ is given in (3.22).} \tag{3.24}$$

Note that when there is infinite capacity (i.e., $U_{i,t} = \infty$) on the amount of overtime each server may have on each day, we then replace θ^* with θ_1 in (3.24). As shown in Lemmas III.3 and III.4, and Proposition 8, the HOOP-BOT (online Algorithm 1) obtains a feasible dual solution but an almost feasible primal solution. To compute the CR, we take care of infeasibility of the primal solution by converting it into a feasible solution. Let f_P and g_D denote the primal and dual objective values obtained by the online Algorithm 1 and $\tilde{f}_P$ denote the objective value of the feasible solution obtained by converting the almost feasible solution of the algorithm, and f^* denote the objective value of an optimal offline policy. By *weak duality*, we have $f^* \leq g_D$, and then we can write the competitive ratio r as follows:

$$r = \frac{\tilde{f}_P}{f^*} \geq \frac{\tilde{f}_P}{g_D} = \frac{\tilde{f}_P}{f_P} \cdot \frac{f_P}{g_D} = \frac{\tilde{f}_P / f_P}{g_D / f_P}. \tag{3.25}$$

We know $g_D / f_P \leq (1 + \theta^*)$ from (3.24), but need to find the *tightest lower bound* for the ratio $\tilde{f}_P / f_P$ by constructing a feasible primal solution from the almost feasible primal solution.

For this purpose, assume $y = \left(y_{i,1,t}^{(1)}, ..., y_{i,m,t}^{(m)}\right); v = \left(v_{i,t}^{(j)}, j = 1 \cdots m\right), \forall i \in [n], t \in [T]$ is an *almost feasible* primal solution obtained by online primal-dual Algorithm 1. We tweak this solution to make a *feasible* primal solution $\tilde{y} = \left(\tilde{y}_{i,1,t}^{(1)}, ..., \tilde{y}_{i,m,t}^{(m)}\right); \tilde{v} = \left(\tilde{v}_{i,t}^{(j)}, j = 1 \cdots m\right), \forall i \in [n], t \in [T]$ where $\tilde{y}_{i,1,t}^{(1)} = y_{i,1,t}^{(1)}$, ..., $\tilde{y}_{i,m,t}^{(m)} = y_{i,m,t}^{(m)}$ but $\tilde{y}_{i,K_1,t}^{(K_1)} \leq y_{i,K_1,t}^{(K_1)}$, and $\tilde{y}_{i,K_2,t}^{(K_2)} \leq y_{i,K_2,t}^{(K_2)}$, and also $\tilde{v}_{i,t}^{(K_2)} \leq v_{i,t}^{(K_2)}$ for $\forall i \in [n], t \in [T]$. Indeed, we obtain a feasible solution $(\tilde{y}, \tilde{v})$ by

only changing the value of $y^{(K_1)}_{i,K_1,t}$, $y^{(K_2)}_{i,K_2,t}$ and $v^{(K_2)}_{i,t}$ of the solution obtained from the online Algorithm 1.

Next, we compute the objective function ratio of the almost feasible primal solution $(\tilde{y}, \tilde{v})$ to the feasible primal solution (y, v) for each server-date (i, t):

$$\frac{\tilde{f}^{i,t}_P}{f^{i,t}_P} = \frac{\sum_{j=1}^{m} c_{i,j}\, \tilde{y}^{(j)}_{i,j,t} - d_i\ \tilde{v}^{(j)}_{i,t}}{\sum_{j=1}^{m} c_{i,j}\, y^{(j)}_{i,j,t} - d_i\ v^{(j)}_{i,t}} \tag{3.26}$$

$$= \frac{\left(\sum_{j=1}^{m} c_{i,j}\, y^{(j)}_{i,j,t} - d_i\ v^{(j)}_{i,t}\right) - c_{i,K_1}\ y^{(K_1)}_{i,K_1,t} + c_{i,K_1}\ \tilde{y}^{(K_1)}_{i,K_1,t} - c_{i,K_2}\ y^{(K_2)}_{i,K_2,t} + c_{i,K_2}\ \tilde{y}^{(K_2)}_{i,K_2,t} + d_i\ b_{i,K_2}}{\left(\sum_{j=1}^{m} c_{i,j}\, y^{(j)}_{i,j,t} - d_i\ v^{(j)}_{i,t}\right)} \tag{3.27}$$

$$= 1 - \frac{c_{i,K_1}\ (y^{(K_1)}_{i,K_1,t} - \tilde{y}^{(K_1)}_{i,K_1,t}) + c_{i,K_2}\ (y^{(K_2)}_{i,K_2,t} - \tilde{y}^{(K_2)}_{i,K_2,t}) - d_i\ b_{i,K_2}}{\left(\sum_{j=1}^{m} c_{i,j}\, y^{(j)}_{i,j,t} - d_i\ v^{(j)}_{i,t}\right)} \tag{3.28}$$

$$\geq 1 - \frac{c_{i,K_1}\ (y^{(K_1)}_{i,K_1,t} - \tilde{y}^{(K_1)}_{i,K_1,t}) + c_{i,K_2}\ (y^{(K_2)}_{i,K_2,t} - \tilde{y}^{(K_2)}_{i,K_2,t})}{\left(\sum_{j=1}^{m} (c_{i,j} - b_{i,j}\ d_i)\, y^{(j)}_{i,j,t}\right)} \tag{3.29}$$

$$= 1 - \frac{\left(\frac{c_{i,K_1}}{b_{i,K_1}}\right)\ b_{i,K_1}\left(y^{(K_1)}_{i,K_1,t} - \tilde{y}^{(K_1)}_{i,K_1,t}\right)}{\sum_{j=1}^{m}\left(\frac{\bar{c}_{i,j}}{b_{i,j}}\right) b_{i,j} y^{(j)}_{i,j,t}} - \frac{\left(\frac{c_{i,K_2}}{b_{i,K_2}}\right)\ b_{i,K_2}\left(y^{(K_2)}_{i,K_2,t} - \tilde{y}^{(K_2)}_{i,K_2,t}\right)}{\sum_{j=1}^{m}\left(\frac{\bar{c}_{i,j}}{b_{i,j}}\right) b_{i,j} y^{(j)}_{i,j,t}} \tag{3.30}$$

$$\geq 1 - \left(\frac{\eta_i}{\beta_i}\right)\left(\frac{b_{i,K_1}\left(y^{(K_1)}_{i,K_1,t} - \tilde{y}^{(K_1)}_{i,K_1,t}\right)}{\sum_{j=1}^{m} b_{i,j}\, y^{(j)}_{i,j,t}} + \frac{b_{i,K_2}\left(y^{(K_2)}_{i,K_2,t} - \tilde{y}^{(K_2)}_{i,K_2,t}\right)}{\sum_{j=1}^{m} b_{i,j}\, y^{(j)}_{i,j,t}}\right) \tag{3.31}$$

where $\eta_i = \max_j \left\{\frac{c_{i,j}}{b_{i,j}}\right\}$.

Since $\left(y^{(K_1)}_{i,K_1,t} - \tilde{y}^{(K_1)}_{i,K_1,t}\right)$ is either 1 if $\sum_{j=1}^{K_1} b_{i,j} y^{(j)}_{i,j,t} > B_{i,t}$ or 0 otherwise, and $\left(y^{(K_2)}_{i,K_2,t} - \tilde{y}^{(K_2)}_{i,K_2,t}\right)$ is either 1 if $\sum_{j=1}^{K_2} b_{i,j} y^{(j)}_{i,j,t} > B_{i,t} + U_{i,t}$ or 0 otherwise, we can write (3.31) as follows:

$$\frac{\tilde{f}^{i,t}_P}{f^{i,t}_P} \geq \min_{i,t}\left\{1 - \left(\frac{\eta_i}{\beta_i}\right)\left(\frac{b_{i,K_1}\left(y^{(K_1)}_{i,K_1,t} - \tilde{y}^{(K_1)}_{i,K_1,t}\right)}{\sum_{j=1}^{m} b_{i,j} y^{(j)}_{i,j,t}} + \frac{b_{i,K_2}\left(y^{(K_2)}_{i,K_2,t} - \tilde{y}^{(K_2)}_{i,K_2,t}\right)}{\sum_{j=1}^{m} b_{i,j} y^{(j)}_{i,j,t}}\right)\right\} = 1 - 2\gamma_{\max} R_{\max} \tag{3.32}$$

where $\gamma_{\max} = \max_i \{\eta_i\}$. Therefore, $\tilde{f}_P/f_P \geq \min_{i,t} \left(\tilde{f}_P^{i,t}/f_P^{i,t}\right) = 1 - 2\gamma_{\max} R_{\max}$. Finally, by using the result of Proposition 6 that $g_D/f_P \leq (1+\theta^*)$, we have

$$r = \frac{\tilde{f}_P}{f^*} \geq \frac{\tilde{f}_P}{g_D} = \frac{\tilde{f}_P}{f_P} \cdot \frac{f_P}{g_D} = \frac{\tilde{f}_P/f_P}{g_D/f_P} = \frac{1 - 2\gamma_{\max} R_{\max}}{1+\theta^*}. \tag{3.33}$$

In particular, when $R_{\max} \to 0$, the coefficient θ^* is simplified to θ^{**} in (3.12), and $r = \frac{1}{1+\theta^{**}}$. □

Remark. When there is no overtime for servers, or $d_i > \gamma_{\max}$ for all $i \in [n]$, then the bound for the objective function ratio of the almost feasible primal solution $(\tilde{y}, \tilde{v})$ to the feasible primal solution (y, v) becomes $\tilde{f}_P/f_P \geq 1 - \gamma_{\max} R_{\max}$ as there is only one almost feasible solution in this case. The constant in the competitive ratio is $\theta^* = \theta_1 = \frac{\zeta}{(\alpha-1)}$, and thus the competitive ratio is $r = \frac{1-\gamma_{\max} R_{\max}}{1+\theta_1} = 1 - 1/e$, which recovers the classical result (without overtime).

3.5 Online Scheduling Problem with a Rolling Horizon Approach

So far, the scheduler has a T-day horizon, within which to schedule incoming customers. A horizon of T implies that all accepted patients are seen within T days. To make this setting practical, we extend the HOOP-BOT (online Algorithm 1) into a setting in which every day we "roll the horizon forward one day" so that at day t, the horizon is modified to $t+T$ days because we add a fresh day to the calendar every day. It can be thought of as creating a nested duplicate of the problem in the next period. If we accept a customer who is referred on day t, it means that we assign her/him to a server on an appointment day within the scheduling horizon $\{t, t+1, \cdots, t+T\}$. However, the rejection decision now becomes a *deferred* decision. This means that if we are not able to accommodate this customer within the scheduling horizon of T days with respect to her/his arrival day, we defer making the scheduling decision until the next day (although deferring is again allowed on the next or any subsequent day), and treat this as a new arrival for the next day.

From a real-world perspective, the medical clinic does not need to call the patient back until a scheduling decision is made for the patient. It is only deferred patients who cannot be promised an appointment date up front; rather they are required to wait for a call-back on the day on which they are accepted, and their appointment date and a server are provided (usually some days into the future). Call-backs are a practical and common method for specialty outpatient care and surgical services. For example, when there is a referral, in many settings, the specialist clinic takes some time to determine the visit appointment dates available and to call the patient back to set up an appointment. In our experience, up front

patient rejections are frowned upon in hospital care networks. So, this approach gives those who would have been rejected on the previous day a greater chance to be assigned/served, which improves throughput. Our numerical study reveals significant benefits from building this rolling horizon feature into the online algorithms, which allows for increased variability buffering of the demand process. The R-HOOP-BOT (Algorithm 2) is the natural rolling horizon extension of the HOOP-BOT. For the connection to real healthcare practice, see §3.6.3.

Algorithm 2 Online Algorithm for Online Scheduling with a Rolling Horizon (R-HOOP-BOT)

Initially set the dual prices $x_{i,t}, v_{i,t} = 0, u_{i,t} = 1$; $\forall i \in [n],$; $t = t_1, t_2, ..., T$.
for each arriving customer j on day t_1 **do**
  Assign customer j to server i^* on day/slot t^* such that $(i^*, t^*) = \underset{i\in[n],t\in[T]}{\arg\max}\{c_{i,j} - b_{i,j} \cdot x_{i,t}\}$.
  if $(c_{i^*,j} - b_{i^*,j} \cdot x_{i^*,t^*}) \geq 0$ **then**
    Set $z_j \leftarrow c_{i^*,j} - b_{i^*,j} \cdot x_{i^*,t^*}$ **and** $y_{i^*,j,t^*} \leftarrow 1$;
    if $0 \leq x_{i^*,t^*} < d_{i^*}$ **then**
      Set $x_{i^*,t^*} \leftarrow x_{i^*,t^*}\left(1 + \frac{b_{i^*,j}}{B_{i^*,t^*}}\right) + \theta_1 \left(\frac{c_{i^*,j}}{B_{i^*,t^*}}\right)$ [*Updating Rule* (I)].
      if $x_{i^*,t^*} > d_{i^*}$ **then set** $x_{i^*,t^*} \leftarrow d_i$ [*Updating Rule* (II)].
    else:
      Set $u_{i^*,t^*} \leftarrow u_{i^*,t^*}\left(1 + \frac{b_{i^*,j}}{B_{i^*,t^*} + U_{i^*,t^*}}\right) + \theta_2 \left(\frac{\bar{c}_{i^*,j}}{B_{i^*,t^*} + U_{i^*,t^*}}\right)$ [*Updating Rule* (III)].
      Set $x_{i^*,t^*} \leftarrow u_{i^*,t^*} + d_{i^*}$ **and** $v_{i^*,t^*} \leftarrow v_{i^*,t^*} + b_{i^*,j}$.
  else:
    Set $y_{i,j,t}, \leftarrow 0, \; z_j \leftarrow 0; \; \forall i \in [n], \; t \in [T]$.
Set $x_{i,T+1} = 0, \; \forall i \in [n]$; $t_1 \leftarrow t_1 + 1$; $T \leftarrow T + 1$, and Go to Step 2.

Proposition 10. The online Algorithm 2 preserves the competitive ratio of Algorithm 1.

Proof. Proof of Proposition 10: The proof is omitted as it is similar to that of Theorem III.1. □

3.6 Case Study: Empirical Results and Practical Insights

We develop a system model and provide numerical results to evaluate the empirical performance of our proposed HOOP-BOT and R-HOOP-BOT (online Algorithms 1 and 2) using real healthcare appointment scheduling data. Our models/algorithms and case study

are intentionally of a general nature so they can apply to either destination medical centers or typical hospitals.

3.6.1 Data Description and Experiment Setup

We create a model that is driven by recent data on appointment visits of patients from a medical clinic of our partner health system. The clinic offers appointment visits to provide diagnosis, consultation, and procedures (e.g., bladder cancer, kidney stones, prostate cancer, micro-surgical urology, kidney cancer). The medical clinic operates five days a week from 8:00 am to 5:00 pm and is staffed with 16 providers: 10 physicians (MDs) and 6 physician assistants (PAs). These providers have different "skills" or "sub-specialties", including (i) oncology (Onco), (ii) endoscopy (Endo), (iii) general, (iv) neurourology and pelvic reconstructive surgery (NPR), and (v) andrology (Andro). Table 3.1 presents the specialty/sub-specialty and availability of each provider at our partner medical clinic on each working day of a week. Furthermore, each patient has a type/class that is based on their chief complaint and/or diagnoses as well as the nature of the visit, which includes new patients, return visit patients, patients for procedure, patients for biopsy, etc.

Monday	**Tuesday**	**Wednesday**	**Thursday**	**Friday**
PA1 (General)	MD4 (NPR)	MD7 (NPR)	PA3 (Endo)	PA6 (Onco)
MD1 (Andro)	MD5 (General)	MD8 (Endo)	PA1 (General)	MD9 (General)
MD2 (Onco)	PA2 (General)	MD9 (General)	PA6 (Onco)	PA4 (NPR)
MD3 (NPR)	PA6 (Onco)	MD2 (Onco)	MD10 (NPR)	PA3 (Endo)
PA5 (NPR)	PA3 (Endo)	PA3 (Endo)	MD9 (General)	MD2 (Onco)

Table 3.1: The availability of 10 physicians (MDs) and 6 physician assistant (PAs) of a medical clinic from our partner health system over five working days of a week along with their specialty/sub-specialties.

Our data set contains about 5021 appointment visits for which we can identify (i) the diagnoses or chief complaint of the patient, (ii) the date that the patient is referred to the medical clinic to make an appointment so that the arrival process is aligned with when requests come in, (iii) the patient's service time, and (iv) the provider (an MD or PA depending on the service type needed) the patient was allocated to. The process for setting urgency rewards is beyond the scope of our paper, but our approach requires the medical clinic to determine the urgency reward vector $\mathbf{c}_j$ for each incoming patient j based on the patient's diagnoses or chief complaint included in the data set. Our formulation allows us to use urgency reward parameters to incentivize shorter waits (access delays) for more urgent patients.

The number of patients who arrive to request an appointment on each day in the med-

ical clinic is shown in Figure 3.5. It can be seen that the arrival pattern of patients has high variability and non-stationarity (particularly toward the end), which implies high uncertainty. This makes it very interesting to investigate the potential benefits of our proposed online appointment scheduling algorithms that do not know anything about the future arrival pattern.

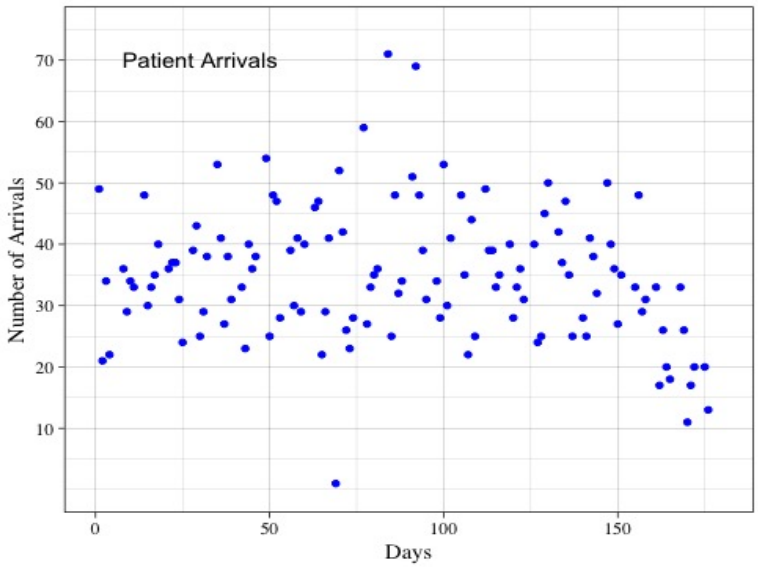


Figure 3.5: The total number of patient arrivals on each day in the medical clinic of our partner health system. The arrival pattern of patients into the medical clinic has highly variability and non-stationarity.

3.6.2 Empirical Performance of the HOOP-BOT

First, we evaluate the empirical performance of the HOOP-BOT (online Algorithm 1) by simulating random instances sampled from the real data and conducting a sensitivity analysis of the CR as service times and overtime costs are varied. Second, we investigate the impact of ignoring stochasticity in service times on the CR of the HOOP-BOT.

Empirical competitive ratio analysis of the HOOP-BOT. This first study focuses only on the oncology service, which provides many urgent service visits within our partner medical clinic. Per the clinic's scheduling guideline, patient requests in the oncology service should be scheduled within 5 days of arrival. We simulate random problem instances for scheduling this specific service type for the HOOP-BOT (online Algorithm 1). For this analysis, we consider a *non-rolling horizon problem* in which the patient requests arrive sequentially on the first day of a 5-day week and must be scheduled within this 5-day scheduling horizon. In contrast, our next study in §3.6.3 will evaluate the R-HOOP-BOT,

where patients may arrive on every day, and the algorithm may choose to defer the unaccepted patients to the next day. Table 3.1 indicates that there are two providers (one MD and one PA) in the medical clinic, serving oncology appointments. To evaluate the empirical performance of the HOOP-BOT, we implement it on a range of test instances with respect to the matrices for service times $\mathbf{b}$ and overtime costs $\mathbf{d}$.

Test type	Performance measure	Overtime cost scenarios $\mathbf{d} = (d_1, d_2)$										
		(0, 1/3)	(0, 2/3)	(0, 1)	(1, 2)	(2, 3)	(2, 4)	(3, 4)	(3, 5)	(4, 5)	(6, 8)	(10, 14)
b	Empirical r	0.956	0.932	0.846	0.924	0.816	0.793	0.763	0.736	0.678	0.668	0.623
	$\frac{\text{Overtime cost}}{\text{Reward}}$	0.156	0.218	0.083	0.078	0.051	0	0	0	0	0	0
1.5**b**	Empirical r	0.915	0.856	0.815	0.783	0.733	0.713	0.607	0.588	0.557	0.549	0.536
	$\frac{\text{Overtime cost}}{\text{Reward}}$	0.248	0.098	0.078	0.088	0.112	0	0	0	0	0	0
2**b**	Empirical r	0.823	0.691	0.721	0.688	0.702	0.654	0.591	0.552	0.539	0.536	0.525
	$\frac{\text{Overtime cost}}{\text{Reward}}$	0.287	0.212	0.152	0.225	0	0	0	0	0	0	0
2.5**b**	Empirical r	0.765	0.717	0.699	0.637	0.635	0.635	0.576	0.549	0.528	0.531	0.515
	$\frac{\text{Overtime cost}}{\text{Reward}}$	0.152	0.119	0	0	0	0	0	0	0	0	0
3**b**	Empirical r	0.705	0.689	0.667	0.627	0.627	0.581	0.527	0.527	0.544	0.512	0.482
	$\frac{\text{Overtime cost}}{\text{Reward}}$	0.078	0	0	0	0	0	0	0	0	0	0

Table 3.2: The empirical competitive ratios and the overtime cost over reward ratios obtained by implementing the HOOP-BOT (online Algorithm 1) for different values of service time $\boldsymbol{b}$ and overtime cost $\boldsymbol{(d_1, d_2)}$ scenarios.

Recall that the urgency/reward value vector $\mathbf{c}_j = (c_{1,j}, \ldots, c_{n,j})$ for each patient j must identify the relative urgency of each service provider assignment. This urgency value is in part determined by the type of visit needed, which is specified by the chief complaint and/or diagnoses included in our data for that patient and requires expert judgment from clinical practice. It can be adjusted to reflect the relative preference of having one provider (e.g., MD in this case) perform the service as opposed to another (e.g., PA in this case). The dependency of each patient's urgency/reward value on different providers is permitted for generality in order to discourage or encourage particular provider-patient pairings/matchings in our online algorithms. The clinic may set the reward to zero for patients who should not be seen by a particular provider while using higher values for the most appropriate providers. It also helps create an appointment scheduling system for a clinic/hospital in which physicians spend more time "practicing at the top of their license." It is worth noting that we can also address "continuity of care" issues in our proposed online policies. For example, out of all the providers qualified to serve a patient, perhaps only 1 or 2 have previously treated the patient and can, therefore, be assigned higher rewards to encourage those pairings/matchings. In our analysis, the service times are set based on the operations of our partner clinic. In particular,

the service times for the various visit types of each provider may vary. The base/nominal service time vectors are $\mathbf{b} = (10, 20, 30, 50)$ for PA6 and $\mathbf{b} = (15, 30, 45, 60)$ for MD2 in minutes in our data set. As is common practice, providers attempt to match the nominal visit lengths (or at least not fall behind schedule), and patient waiting and overtime occur when the appointment visits take longer than planned for.

One goal is to evaluate the empirical performance of the proposed *online policy* by the HOOP-BOT in comparison to the *optimal offline policy*. Further, we wish to gain insight into the sensitivity of service time lengths and overtime penalties, so we conduct a sensitivity analysis for the impact of service time and overtime cost on the CR of the proposed online policy. To this aim, we scale each provider's nominal $\mathbf{b}$ uniformly by the factors of 1.5, 2, 2.5 and 3 to create five different test instances (see "Test type" column in Table 3.2). Furthermore, we consider eleven scenarios for the overtime cost of the PA6 and MD2 (see "Overtime cost scenarios (d_1, d_2)" columns in Table 3.2). We report (i) the empirical competitive ratio r and (ii) the ratio of overtime cost over the total reward as the two performance measures of the HOOP-BOT for each instance in Table 3.2. Note that we choose not to conduct a sensitivity analysis on the urgency rewards $\mathbf{c}$, because they must be fixed numbers for each patient and chosen carefully by the medical clinic to induce the right prioritization of patients. The ranges are $c \sim \text{uniform}[120, 155]$ for emergent patients, $c \sim \text{uniform}[45, 95]$ for urgent patients, and $c \sim \text{uniform}[10, 25]$ for elective patients.

We have several observations and insights from the computational results in Table 3.2. First, our proposed online policies perform significantly better than the theoretical performance guarantee and are close to optimal in many cases. Note that the optimal offline solution used in the competitive analysis is not achievable in reality, so our performance is even better than these numbers suggest. Second, when the per-unit-time overtime cost is low, the online policies perform near optimally, but as the per-unit-time overtime cost increases, the empirical CR decreases. This is also consistent with our theoretical results. In fact, as overtime cost increases, there is less effective capacity, and so the optimal offline policy has an increased advantage because it can achieve careful "bin-packing." Third, when the service time requirement is short, the online policies perform near optimally. As the service time requirement increases, the empirical CR decreases. This insight is not only consistent with our theoretical results but also intuitive in the sense that when we increase the service requirement, our online algorithms will have less flexibility in making the allocation and overtime decisions, resulting in a lower CR.

Impact of stochasticity in service time on competitive ratio. One of the assumptions in the OS-BOAA problem is that the service time requirements of patients are deterministic (see §3.2.1). Also, our theoretical performance guarantees do not consider

stochastic service times. To evaluate the impact of ignoring likely stochasticity in service times, we apply a *certainty equivalent* approach in which the scheduling decisions made by the HOOP-BOT using mean service times and performance is evaluated on stochastic service times.

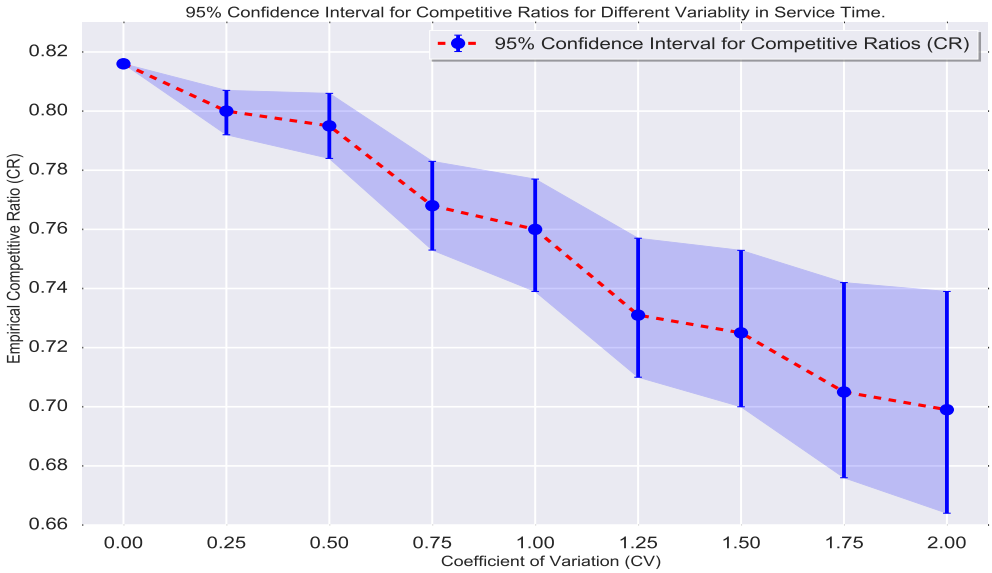


Figure 3.6: Illustration of impact of ignoring stochasticity in service time on CR of the HOOP-BOT: %95 confidence intervals obtained for CRs of 1000 sample paths of stochastic service times corresponding to each cv. The blue circles are the mean of CRs and the CR for $cv = 0$ is obtained for the deterministic service times.

To this aim, we consider $\mathbf{b} + \epsilon$ where $\mathbf{b}$ is the base service time, and $\epsilon \sim N(0, \sigma(\epsilon))$ is a random noise with $\sigma(\epsilon) = cv \times \mathbf{b}$, and cv is the coefficient of variation. We generate 1000 sample paths for the stochastic service times of patients for each value of $cv \in \{\frac{1}{4}, \frac{1}{2}, \frac{3}{4}, 1, 1\frac{1}{4}, 1\frac{1}{2}, 1\frac{3}{4}, 2\}$, simulate the online policy derived by the HOOP-BOT (online Algorithm 1) on these sample paths, and calculate a 95% confidence interval for the resulting CRs corresponding to each cv value. Note that to calculate the CR, the optimal offline policy is obtained by solving the *expected* primal problem (P) where the expected service times are used. According to Jensen's inequality, this results in an upper bound for the CR.

Figure 3.6 illustrates the impact of ignoring stochasticity in service times on the CR of the HOOP-BOT for varying levels of service time variability. Note that the CR for $cv = 0$ is for the case of deterministic service times. The first insight is that even though the CRs become worse in the case of stochastic service times, the proposed online policy is fairly robust with respect to stochasticity in service time. Second, as the value of cv increases, we have more variability in the competitive ratio of the online policy.

3.6.3 Empirical Performance of the R-HOOP-BOT (Rolling Horizon)

First, we assess the empirical performance of the R-HOOP-BOT (Algorithm 2) by comparing it with two benchmarks on real data from our partner health system. Second, we investigate the effect that a rolling horizon framework and deferring scheduling decisions have on the amount of delay in serving patients for several proposed policies.

Assessing performance of different scheduling policies in healthcare practice. We analyze and evaluate our proposed online policy and some benchmark policies as follows.

- Online Policy: This is our proposed online policy obtained by the R-HOOP-BOT with a *5-day-rolling horizon*.
- First-Come First-Served Policy (FCFS): This is a pervasive and easy-to-implement policy for many service systems. In this policy, every incoming patient is allocated to the server with the earliest availability within the scheduling horizon as long as his/her requirements can be met with remaining capacity of that provider (which is the sum of regular and overtime capacity on each day). This policy is blind to urgency.
- Nested Threshold Policy with Overtime (NTPO): We have designed this new policy, which is based on the idea of protecting more urgent provider-patient pairs with some pre-specified thresholds to reserve capacity a priori, and also serving them with overtime if necessary afterward. This idea is drawn from the airline revenue management. The details follow.

Clearly, the FCFS policy matches every incoming patient to the first available server-date and does not consider any reasonable match between the patient and the server. So, we compare our online algorithms with a more sensible class of policies to evaluate better the performance of our proposed online framework. We shall call it *Nested Threshold Policy with Overtime* (NTPO) and construct it as follows. Let c^{UB} and c^{LB} be upper and lower quantities for the possible values of the reward/urgency $c_{i,j}$. We uniformly partition the range $[c^{LB}, c^{UB}]$ of possible urgencies into three equal sections by using two *reward/urgency thresholds* c_1 and c_2 to create three urgency classes. These classes are the (i) *emergent* class or class 1 if $c_2 \leq c_{i,j} \leq c^{UB}$, (ii) *urgent* class or class 2 if $c_1 \leq c_{i,j} \leq c_2$, and (iii) *elective* class or class 3 if $c^{LB} \leq c_{i,j} \leq c_1$. Let $k(i,j)$ denote the class of patient j if served by provider i. The class of a patient could be different for each server, which enables the incorporation of practicing at the top of license and continuity of care issues.

We divide the regular capacity $B_{i,t}$ and overtime $U_{i,t}$ of each server-date (i,t) into three *capacity buckets* by defining two *protection levels* ρ_1 and ρ_2, and reserve a percentage of the

capacity for each of three buckets. Class 1 patients can exploit all three buckets, class 2 patients can be served by buckets 2 and 3, and class 3 patients can be served only by bucket 3 of each server. The priorities of buckets are decreasing from bucket 1 to bucket 3 for class 1 patients, so they are first allocated to the reserved capacity of bucket 1 if possible. If there is no capacity in bucket 1, then reserved bucket 2 capacity is used and, if the reserved bucket 2 capacity is used up, then the class 1 patient is served from bucket 3. If even the overtime capacity of bucket 3 is used up, then the patient is deferred, and a slot is allocated on the earliest day at which capacity can be found. We have the analogous priority rule for serving class 2 patients by buckets 2 and 3. The other indices and parameters of the NTPO are given in Table 3.3.

Parameters of the Nested Threshold Policy with Overtime (NTPO).	
b	: The index $b \in \{1,2,3\}$ denotes the capacity bucket of each server-date (i,t).
$k(i,j)$	: The index $k(i,j) \in \{1,2,3\}$ denotes the class of a patient j if served by server i.
$B^b_{i,t}$	: The remaining regular capacity of server-date (i,t) in bucket b.
$U^b_{i,t}$	: The remaining overtime capacity of server-date (i,t) in bucket b.
$L^b(i,t,j)$	: Binary parameter equal to 1 if $b \geq k(i,j)$ and $B^b_{i,t} \geq b_{i,j}$, and 0 otherwise.
$Q^b(i,t,j)$	: Binary parameter equal to 1 if $b \geq k(i,j)$ and $U^b_{i,t} \geq b_{i,j}$, and 0 otherwise.

Table 3.3: The definitions of the parameters and indices used in the nested threshold policy with overtime.

The binary parameters $L^b(i,t,j)$ and $Q^b(i,t,j)$ determine whether (i) server-date (i,t) has enough regular or overtime capacity in bucket b to serve patient j, respectively, and also (ii) bucket index b is greater than or equal to the class index $k(i,j)$ of patient j. Consequently, we define the two sets $\mathcal{M}_j = \{(i,t) \mid \sum_{b=k(i,j)}^{3} L^b(i,t,j) \geq 1, t \in T\}$ and $\mathcal{N}_j = \{(i,t) \mid \sum_{b=k(i,j)}^{3} Q^b(i,t,j) \geq 1, t \in T\}$ for each incoming patient j. These are the set of all servers that can serve patient j in one of their regular or overtime buckets, respectively. Algorithm 3 presents the NTPO for the OS-BOAA problem.

Before comparing our online policies with the above benchmarks, we need to carefully adjust the protection levels ρ_1 and ρ_2 for the NTPO. We conduct a sensitivity analysis on different pairs of ρ_1 and ρ_2 to find which pair maximizes total reward in the class of all nested threshold policies with overtime. As reported in Table 3.4, $(\rho_1, \rho_2) = (0.4, 0.2)$ are the best protection levels for our case study. Note that the urgency/reward thresholds c_1 and c_2 are set by the medical clinic, but for our reward model described in §3.6.2, $(c_1, c_2) = (100, 30)$ were used.

We consider the *5-day-rolling horizon* version of the OS-BOAA problem and compare the empirical and theoretical performance of our online policies obtained by the R-HOOP-BOT (online Algorithm 2), with the ones of FCFS and NTPO benchmarks using a 5-day horizon that is rolled forward for 20 days. All these scheduling policies set a visit date and a provider

Algorithm 3 Nested Threshold Policy with Overtime (NTPO).

Initially set reward thresholds c_1 and c_2, and protection levels ρ_1 and ρ_2.
for each arriving customer j on day t_1 **do**
    if $\mathcal{M}_j \cup \mathcal{N}_j \neq \varnothing$ **then**
        Choose $(\tilde{i}, \tilde{t})$ with the earliest availability from the set $\mathcal{M}_j \cup \mathcal{N}_j$.
        if $\sum_{b=k(\tilde{i},j)}^{3} L^b(\tilde{i}, \tilde{t}, j) \geq 1$ **then**
            Allocate j to $\tilde{b} = \arg\min_{b \in \{k(\tilde{i},j),\ldots,3\}} \{\, b \mid L^b(\tilde{i}, \tilde{t}, j) = 1\}$ of $(\tilde{i}, \tilde{t})$.
        else:
            Allocate j to $\tilde{b} = \arg\min_{b \in \{k(\tilde{i},j),\ldots,3\}} \{\, b \mid Q^b(\tilde{i}, \tilde{t}, j) = 1\}$ of $(\tilde{i}, \tilde{t})$.
        Update remaining capacity $B^b_{\tilde{i},\tilde{t}}$ and overtime capacity $U^b_{\tilde{i},\tilde{t}}$ of $(\tilde{i}, \tilde{t})$.
    else:
        Do not allocate patient j and defer decision for patient j into next day.
Set $t_1 \leftarrow t_1 + 1$; $T \leftarrow T + 1$, and Go to Step 2.

(ρ_1,ρ_2)	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
0.1	45,312	51,521	35,125	60,202	41,425	39,234	26,124	45,148	32,215
0.2	35,451	42,119	55,102	67,325	38,423	28,961	53,226	48,156	-
0.3	65,934	72,452	78,325	80,923	75,125	71,568	69,328	-	-
0.4	78,436	92,974	87,624	82,124	75,875	68,245	-	-	-
0.5	74,265	89,911	78,218	67,327	59,145	-	-	-	-
0.6	64,281	80,127	75,347	79,126	-	-	-	-	-
0.7	65,111	57,122	60,217	-	-	-	-	-	-
0.8	48,272	50,347	-	-	-	-	-	-	-
0.9	27,546	-	-	-	-	-	-	-	-

Table 3.4: Finding the best set of protection levels ρ_1 and ρ_2 for the nested threshold policy with overtime by implementing the NTPO (Algorithm 3) on the real data from our partner medical clinic, and calculating the total objective function for each pair. Numerically, (0.4, 0.2) are the best protection levels ρ_1 and ρ_2 in our case study.

Overtime Cost	**d**	**2d**	**3d**	**4d**	**5d**	**6d**	**7d**
Avg Empirical r	0.932	0.894	0.742	0.792	0.695	0.652	0.636
Theoretical r	0.634	0.535	0.463	0.408	0.365	0.329	0.301
Empirical r of NTPO	0.573	0.517	0.429	0.382	0.335	0.301	0.276
Empirical r of FCFS	0.342	0.356	0.332	0.297	0.253	0.262	0.231

Table 3.5: Empirical Performance Evaluation of the Online Policy: The average empirical and theoretical competitive ratios of the online policies obtained by the R-HOOP-BOT (online Algorithm 2) and their comparison with the empirical competitive ratios of FCFS and NTPO policies on the real appointment-scheduling data from the partner medical clinic for different values of per-unit-time overtime costs.

for each patient and make overtime decisions for each provider on each date while deferring patients that cannot be served within regular capacity or overtime. Recall that rejecting a patient is not an option. While some patients that are deferred will eventually be served, some will not be served within the horizon (25 days in our experiment), which may be thought of as a form of rejection. We report the average empirical CRs of all these policies, along with the theoretical CR of the online policy for different per-unit-time overtime costs in Table 3.5. The average empirical CRs are obtained by taking the average of CRs of all the days over 20 days of "rolling" forward the problem. That is, each day of the week was captured in a separate problem with the day's arrivals coming on the first day with the four remaining days in a week to serve the arrivals. These are connected as the horizon rolls forward one day at a time beginning from the prior day's state.

The computational results in Table 3.5 suggest that the online policy obtained by the R-HOOP-BOT (Algorithm 2) not only typically performs close to the optimal offline policy, but also outperforms the two benchmarks of the FCFS and NTPO by a large margin on our data set. Compared to FCFS and NTPO, the average percentage improvements of the proposed online policy over the seven cases are 160% and 96%, respectively, in terms of CR. We explain the poor performance of FCFS as occurring because it spends too much time serving non-urgent visits and leaves urgent cases undone at the end of the horizon. Even in the NTPO, which cares about the class of patients and accordingly reserves some part of regular and overtime budgets for each class a priori, the empirical performance is worse than both empirical and theoretical performances of our online policies. The good performance of the online policy can be explained as a result of paying close attention to the urgency/reward value of each arriving patient, computing the dual price of every server-day pair for this patient and then optimizing the server-date selection.

Analysis of patient deferment with different urgencies. We compare the perfor-

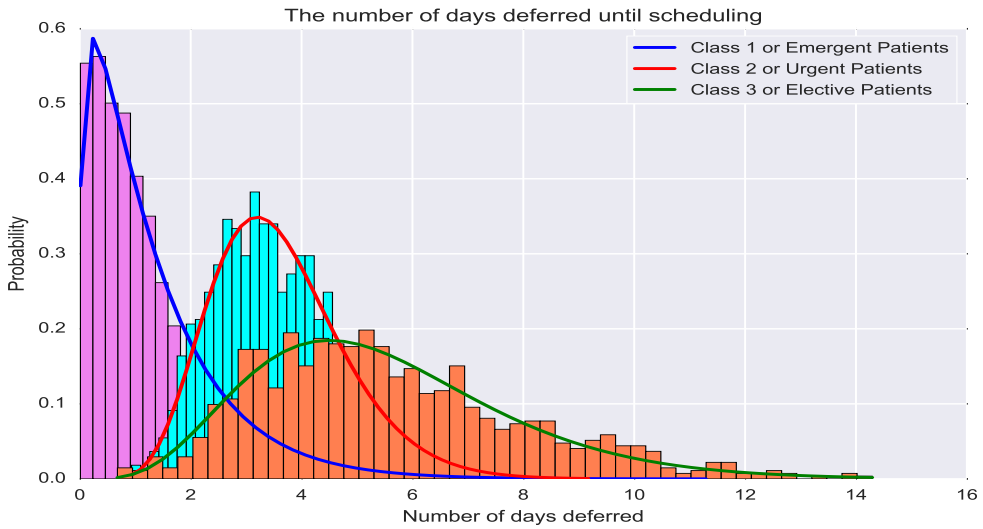


Figure 3.7: The histogram and fitted distributions on the number of days patients in each of three urgency classes are deferred until an allocation decision is made by the online policy obtained by the R-HOOP-BOT on the real data.

mance of the online policy obtained by the R-HOOP-BOT with the NTPO in terms of the number of days patients are deferred, as well as the percentage of rejections for different urgency classes of patients in the rolling horizon framework. We use the online policy and the NTPO on the real data from our partner medical clinic, and graph the number of days patients in each of three urgency classes are deferred until an admissible allocation date is found.

Performance Measure	Online Policy			NTPO Policy		
	Class 1	Class 2	Class 3	Class 1	Class 2	Class 3
Mean	1.131	3.612	5.521	1.211	7.752	10.921
Std. Deviation	1.204	1.212	2.345	1.018	2.364	4.453
Percentage of Rejection	1.251%	2.421%	5.261%	4.432%	9.612%	12.142%

Table 3.6: Empirical Performance Evaluation of the Online Policy: (i) the mean, and standard deviation for the number of days patients are deferred until an allocation decision is made, and (ii) the percentage of rejection decisions made by the online policy and NTPO policy for each of three urgency classes.

Figures 3.7 and 3.8 illustrate the histograms and fitted distributions on the number of days patients are deferred in each of these classes, and Table 3.6 reports the related statistics for these results. These histograms report fractions of days because we used actual arrival times of patients in the data in our computations. While the online policy has no classes,

we sort the patients according to the classes as defined above to form classes for the NTPO policy. We observe that (i) the online policy obtained by the R-HOOP-BOT outperforms the NTPO in terms of both mean and standard deviation for the number of deferred days (with very similar results for class 1), and (ii) it offers a much lower percentage of rejections, which implies higher throughput. Note that in this analysis, we ignore the result of the FCFS policy, because it does not distinguish between urgency classes while making the scheduling decisions, so all have a similar access delay distribution.

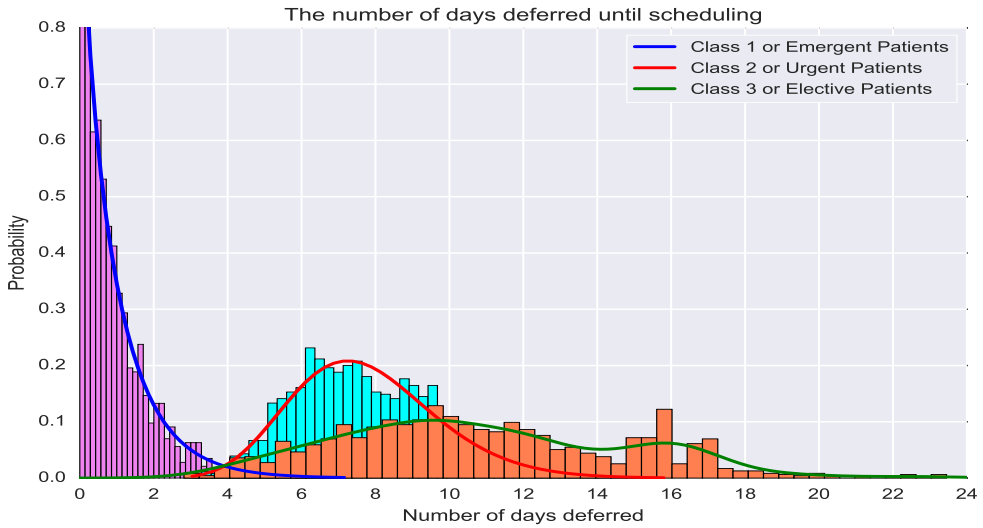


Figure 3.8: The histogram and fitted distributions on the number of days patients in each of three urgency classes are deferred until an allocation decision is made for them by the NTPO policy on the real data. Note that the x axis is much longer than the one in Figure 3.7.

3.7 Conclusion and Future Directions

In this paper, we study an important class of online scheduling problem with budgeted overtime in which the model has no knowledge about the pattern or underlying distribution of the arrival process. Upon arrival of a customer, the system makes an instantaneous and irrevocable allocation decision for this customer, without knowing any information on subsequent customers. We adopt a primal-dual approach to develop new effective and efficient online algorithms to make for every arriving patient on every day in the horizon not only a date-server allocation decision but also a decision of whether or not to use overtime to serve the patient. The proposed online policies (i) are robust to future uncertain information, (ii) are easy to implement and extremely efficient to compute, (iii) allow for heterogeneity in both reward and service requirement by a server, and (iv) admit a theoretical performance

guarantee. Comparing our online policy with the optimal offline policy, we obtain a competitive ratio which guarantees the worst-case performance of our proposed online policy. For practical settings, we extend our online algorithm to a rolling horizon paradigm.

A particular emphasis of this paper has been put on the real-world applicability of our proposed methods. The online resource allocation problem studied in this work is not only investigated through a theoretical lens but also from the perspective of healthcare operations. We evaluate the empirical performance of our online algorithms by using real appointment-scheduling data from a healthcare clinic of our partner health system. Our computational results show that the proposed online policies perform much better than their theoretical worst-case performance guarantee and extremely well compared to the pervasive FCFS scheduling heuristic and a new policy we term the nested threshold policy.

There are several limitations in our current model that could spur future research. First, we do not consider stochasticity in the service time requirement in our theoretical analysis (even though we investigate it empirically). Thus, it would be interesting to see if one can design an online algorithm that can handle stochastic service times and obtain a CR. Second, the rolling horizon extension is myopic and not optimal. It would be interesting to study the full-blown dynamic optimization problem and establish meaningful theoretical results. Third, in practice, patient no-shows and cancellations happen from time to time. No-shows do not impact the model at the daily level (only time of day). The current methodology does not incorporate cancellations, and we leave it for future research. Lastly, there is a small gap between our lower and upper bounds for the competitive ratio of our online algorithms, so it is not tight. Thus, whether the proposed online primal-dual algorithms admit a tighter lower bound or there is a tighter upper bound for any online algorithm remains a question for future research.

3.8 Appendix

3.8.1 Appendix A: Summary of Major Notation

Table 3.7 summarizes the major mathematical notation used in the manuscript.

Notation	Description
i	index of servers/providers.
t	index of periods/days.
j	index of customers/patients.
$b_{i,j}$	service time of customer j if served by server i.
$c_{i,j}$	reward or urgency value of customer j if served by server i.
$B_{i,t}$	regular capacity of server i on day t.
$U_{i,t}$	overtime capacity of server i on day t.
d_i	overtime cost of server i.
$y_{i,j,t}$	binary decision for allocating customer j to server i on day t.
$v_{i,t}$	overtime decision for server i on day t.
$\gamma_{\max} = \max_{i,j}\left\{\frac{c_{i,j}}{b_{i,j}}\right\}$	max ratio of reward to service requirement among all i and j.
$\gamma_{\min} = \min_{i,j}\left\{\frac{c_{i,j}}{b_{i,j}}\right\}$	min ratio of reward to service requirement among all i and j.
$d_{\max} = \max_i\{d_i\}$	max overtime cost among all i.
$d_{\min} = \min_i\{d_i\}$	min overtime cost among all i.
$R_{\max} = \max_{i,j,t}\left\{\frac{b_{i,j}}{B_{i,t}}\right\}$	max ratio of service requirement per request to total capacity.
$\alpha = (1+R_{\max})^{1/R_{\max}}$	coefficient in the competitive ratio.

Table 3.7: Summary of major notation in the OS-BOAA problem and the competitive analysis.

3.8.2 Appendix B: An Upper Bound of Online Algorithms for the OS-BOAA problem

We first prove a technical result that will be used in the proof of Theorem 2.

Lemma III.5. The following inequality holds true.

$$1-\frac{5}{N} \le \sum_{j=1}^{q}\left(\frac{1}{N+1-j}\right) \le 1, \quad \text{where} \quad q = N - \left\lceil \frac{N}{e} \right\rceil. \tag{3.34}$$

Proof. Proof of Lemma III.5. To prove (3.34), we use the integral test for estimating sum of a decreasing series as follows:

$$\sum_{j=1}^{q}\Big(\frac{1}{N+1-j}\Big)=\sum_{x=1+\lceil N/e\rceil}^{N}\frac{1}{x}\;\leq\;\int_{N/e}^{N}\frac{1}{x}\,dx=1.$$

$$\frac{5}{N}+\sum_{j=1}^{q}\Big(\frac{1}{N+1-j}\Big)>\frac{1}{N+5}+\frac{1}{N+4}+\cdots+\frac{1}{N+1}+\sum_{j=1}^{q}\Big(\frac{1}{N+1-j}\Big)$$

$$\geq\sum_{x=2+\lceil N/e\rceil}^{N+6}\frac{1}{x}\;\geq\;\int_{2+N/e}^{N+6}\frac{1}{x}\,dx\;>\;\int_{\frac{(N+6)}{e}}^{N+6}\frac{1}{x}\,dx=1.$$

This completes the proof. □

Proof. Proof of Theorem 2: According to Yao's minimax principle, the expected revenue of a randomized online algorithm on the worst-case arrival input is no better than the one for a worst-case probability distribution over the arrival inputs, of the deterministic online algorithm that performs best against that distribution. Thus, to derive an upper bound r on the performance of randomized online algorithms, it suffices to provide a distribution over worst-case inputs such that any deterministic online algorithm achieves at most r of the optimal offline policy in expectation. Consider an arbitrary deterministic online algorithm, denoted ALG, and evaluate its performance on the worst-case input generated as follows.

Assume that the set of resources $\mathcal{R}=\big\{(i,t)\mid i\in[n],t\in[T]\big\}$ have $N=n\times T$ resources, and each has both regular and overtime capacities. We can think of $2N$ resources including N resources denoted by $r_1,r_3,...,r_K$ with only regular capacity of $B_k=1$ for $k\in\{1,3,...,K\}$, and N resources denoted by $\hat{r}_2,\hat{r}_4,...,\hat{r}_L$ with only overtime capacity of $U_k=1/2$ for $k\in\{2,4,...,L\}$ such that $L+K=2N$. Next, consider all inputs derived from the above input by taking random permutation of the $2N$ resources, so taking a uniform distribution u over N regular resources, and a uniform distribution $\tilde{u}$ over N overtime resources. Formally, with $\sigma(.)$ denoting a permuted order, pick a random permutation $\sigma(r_1),\sigma(r_3),...,\sigma(r_K)$ of N regular resources and a random permutation $\tilde{\sigma}(\hat{r}_2),\tilde{\sigma}(\hat{r}_4),...,\tilde{\sigma}(\hat{r}_L)$ of N overtime resources.

Patients arrive in $2N$ iterations, with $1/\epsilon$ number of patients in each iteration. Let $\mathcal{S}_j$ denote the set of patients at iteration $j\in\{1,2,...,2N\}$. The patients arrive in iterations in the order $\mathcal{S}_1,\mathcal{S}_2,...,\mathcal{S}_{2N}$. For patients $\mathcal{S}_j$, where $j\in\{1,3,...,K\}$, resources $\sigma(r_1),\tilde{\sigma}(\hat{r}_2),\sigma(r_3),...,\tilde{\sigma}(\hat{r}_{j-1})$ and $\tilde{\sigma}(\hat{r}_{j+1}),\tilde{\sigma}(\hat{r}_{j+3}),...,\tilde{\sigma}(\hat{r}_L)$ have reward 0, but resources $\sigma(r_j)$, $\sigma(r_{j+2}),...,\sigma(r_K)$ have reward ϵ. However, for patients $\mathcal{S}_j$, where $j\in\{2,4,...,L\}$, resources $\sigma(r_1),\tilde{\sigma}(\hat{r}_2),\sigma(r_3),...,\tilde{\sigma}(\hat{r}_{j-1})$ and $\sigma(r_{j+1}),\sigma(r_{j+3}),...,\sigma(r_K)$ have reward 0, but resources

$\tilde{\sigma}(\hat{r}_j), \tilde{\sigma}(\hat{r}_{j+2}), ..., \tilde{\sigma}(\hat{r}_L)$ have reward $\epsilon/2$.

Without loss of generality, let us index patients served by regular resources by $j \in \{1, ..., N\}$, patients served by overtime resources by $j \in \{N+1, ..., 2N\}$, resources with regular capacities by $k \in \{1, ..., N\}$, and resources with overtime capacities by $k \in \{N+1, ..., 2N\}$. Clearly, the allocation of the optimal offline policy for any permutation σ and $\tilde{\sigma}$ yields the total reward $OPT = \frac{3N}{2}$ by allocating patients $\mathcal{S}_j$ to resource $\sigma(j)$ if $1 \leq j \leq N$, and resource $\tilde{\sigma}(j)$ if $N+1 \leq j \leq 2N$. The goal is to bound the expected total revenue of the deterministic algorithm ALG over inputs from distributions u and $\tilde{u}$. Let $Z_{j,k}$ denote the fraction of patients from $\mathcal{S}_j$ that is allocated to resource k. Then, we have the following:

$$\mathbb{E}_{\sigma}(Z_{j,\sigma(k)}) = \begin{cases} \frac{1}{N+1-j}, & \text{if } j \leq k \leq N \\ 0, & \text{if } k < j \text{ or } k \geq N+1 \end{cases} \tag{3.35}$$

for patients in iterations $j \in \{1, ..., N\}$, and

$$\mathbb{E}_{\tilde{\sigma}}(Z_{j,\tilde{\sigma}(k)}) = \begin{cases} \frac{1}{2N+1-j}, & \text{if } j \leq k \leq 2N \\ 0, & \text{if } k < j \end{cases} \tag{3.36}$$

for patients in iterations $j \in \{N+1, ..., 2N\}$. Thus, the expected revenue obtained by each resource at the end is either $\min\left\{1, \sum_{j=1}^{k}\left(\frac{1}{N+1-j}\right)\right\}$ for resources $k \in \{1, ..., N\}$ or $\min\left\{\frac{1}{2}, \frac{1}{2}\sum_{j=N+1}^{k}\left(\frac{1}{2N+1-j}\right)\right\}$ for resources $k \in \{N+1, ..., 2N\}$.

Using (3.35) and (3.36), the expected total revenue of all regular and overtime resources obtained by the ALG can be bounded as follows:

$$\begin{aligned} ALG &= \sum_{k=1}^{N} \mathbb{E}_{\sigma}\left[\sum_{j=1}^{k} Z_{j,\sigma(k)}\right] + \sum_{k=N+1}^{2N} \mathbb{E}_{\tilde{\sigma}}\left[\sum_{j=1}^{k} Z_{j,\tilde{\sigma}(k)}\right] \\ &\leq \sum_{k=1}^{N} \min\left\{1, \sum_{j=1}^{k}\left(\frac{1}{N+1-j}\right)\right\} + \sum_{k=N+1}^{2N} \min\left\{\frac{1}{2}, \frac{1}{2}\sum_{j=N+1}^{k}\left(\frac{1}{2N+1-j}\right)\right\}. \end{aligned} \tag{3.37}$$

We first bound the first statement in (3.37) by using the proven technical result (3.34) as follows:

$$\begin{aligned} \sum_{k=1}^{N} \min\left\{1, \sum_{j=1}^{k}\left(\frac{1}{N+1-j}\right)\right\} &\leq \sum_{k=1}^{q}\sum_{j=1}^{k}\frac{1}{N+1-j} + \sum_{k=q+1}^{N} 1 \\ &< \sum_{k=1}^{q}\sum_{j=1}^{k}\frac{1}{N+1-j} + \sum_{k=q+1}^{N}\left(\frac{5}{N} + \sum_{j=1}^{q}\frac{1}{N+1-j}\right) \end{aligned}$$

$$= \sum_{k=q+1}^{N} \frac{5}{N} + \sum_{k=1}^{q} \sum_{j=1}^{k} \frac{1}{N+1-j} + \sum_{k=q+1}^{N} \sum_{j=1}^{q} \frac{1}{N+1-j}$$

$$< 5 + \sum_{j=1}^{q} \frac{q-j+1}{N+1-j} + (N-q) \sum_{j=1}^{q} \frac{1}{N+1-j}$$

$$= 5 + q < 5 + \left(1 - \frac{1}{e}\right) N.$$

Using the same argument but with $q' = 2N - \left\lceil \frac{N}{e} \right\rceil$, we can bound the second expression in (3.37), which results in an upper-bound of $5 + \left(1 - \frac{1}{e}\right) \frac{N}{2}$. As the expected total revenue of the optimal offline policy is $OPT = \frac{3N}{2}$, and the expected total revenue of the ALG is bounded above by $10 + \left(1 - \frac{1}{e}\right) \frac{3N}{2}$ when both regular and overtime resources are considered, so ALG cannot be r-competitive for any $r < \frac{e}{e-1}$. □

CHAPTER IV

Coordinated and Priority-based Surgical Care: An Integrated Distributionally Robust Stochastic Optimization Approach [1]

4.1 Introduction

In a typical service system, it is inevitable that waiting times or delays will be experienced by the customers/clients because of the inherent uncertainty in both arrival processes and service times. In healthcare settings, a long waiting time to receive care is not only an annoyance, but it can also deteriorate health outcomes due to adverse events and increase healthcare costs because of the potential need for additional/more complicated procedures [109, 54, 133]. *Timely access to care* is an essential feature of any high-quality and modern healthcare delivery system [93]. We define "access delay" (or access to care) as the number of days between the day a patient's appointment request/referral is received by a medical center and his/her appointment day with a provider. Access delay can be mitigated by efficiently matching the available resource capacity to patient demand. This is, however, challenging given the inherent and various sources of uncertainty within any healthcare delivery system [124]. Currently, the U.S. is experiencing an increase in demand for medical care due to an aging and growing population, which is outpacing the growth of healthcare providers [121]. This limited capacity along with sharply increasing demand leads to barriers to adequate access to care, and also highlights the importance of efficient utilization of resources, including providers and operating rooms. In this context, *coordination of patient care* throughout the course of treatment and across various clinic and surgery visits helps ensure that patients receive appropriate follow-up treatments without enduring long waiting

[1] Under Revision at Production and Operations Management as Keyvanshokooh, E., Fattahi, M., Kazemian, P., Van Oyen, M. P. (2020), Coordinated and Priority-based Surgical Care: An Integrated Distributionally Robust Stochastic Optimization Approach.

times that can undermine their health condition.

This research is motivated by our collaborations with multiple healthcare institutions that desire to achieve timely access to surgery in their specialized surgical units. Patients with various acuity levels are referred to these surgical units either by their primary care physicians or by other hospital units. These patients first require a clinic consultation appointment with a surgeon, which then may need to be followed by a surgical procedure in the operating room. The decision of whether a patient requires a surgery is made during the patient's clinic consultation visit along with details of the surgery. In this paper, we develop a new optimization-based approach to coordinate clinic and surgery appointment scheduling for such surgical units such that all patients with various acuity levels can be offered a clinic consultation visit and a surgical time (if surgery is needed) within a pre-defined target time window that is clinically safe for them to wait using the minimum overtime possible. We call this the *Coordinated clinic and surgery Appointment Scheduling* (CAS) problem.

It is also worth noting that variability in *appointment request arrival numbers* and *surgery durations* can cause excessive patient waiting times and poor utilization of healthcare resources or high overtime. Unlike prior research that assumes the probability distribution of surgery duration is known (e.g., [55], [56], [63] and [60]), our contribution is to consider how distributional robustness can be achieved using a model where only marginal information including mean, variance and range on surgery duration is used. Creating an accurate probability distribution for surgery duration, which can depend on the surgery type as well as the surgeon performing the surgery, requires a large amount of historical data. In many healthcare settings, however, a wide range of surgeries, limited numbers of cases of each type, and surgeons changing over time result in insufficient historical data to accurately estimate surgery duration distributions for each combination of surgery-surgeon type. Further, it might be impossible to fit distributions tailored to the surgeon and the surgery type for some of the less common procedures. For example, as reported by [115], for approximately half of scheduled cases in the U.S. on any weekday, only five or fewer cases of the same surgery type (narrowly defined) and by the same surgeon have been performed. This motivates our interest in a *robust* scheduling policy that could perform relatively well against a class of surgery duration distributions satisfying only the above described moment (marginal) information in the CAS problem. This paper also incorporates the more traditional Poisson arrival process model. We assume there is usually enough historical data on appointment requests from which stochastic scenarios for the number of patient appointment requests can be made [63].

Methodologically, we advance the literature by integrating the multi-stage stochastic programming and distributionally robust optimization approaches such that the uncertainty in

the number of patient appointment requests/referrals and surgery durations are modeled by a *scenario tree* and a *moment-based ambiguity set*, respectively. We call this new approach the *Integrated Multi-stage Stochastic and Distributionally Robust Optimization* (IMSDRO). The IMSDRO approach (i) specifies the optimal clinic date, (ii) determines the optimal surgery date with the same surgeon who performed the clinic visit (given surgery is needed), and (iii) minimizes and balances the clinic and surgery overtimes of surgeons while incorporating the uncertainty in appointment request arrival and surgery durations. The IMSDRO approach *guarantees* that the pre-defined priority-based clinical and surgical access delay targets are met for all patients. Clinical and surgical overtimes are used, as needed, to achieve these predefined priority-based access delay targets.

The scope of this paper is limited to the following two main aims. (i) We develop an optimization-based model to address the type of coordinated care problem with hard access delay constraints (by optimizing overtime) in a dynamic scheduling manner that allows for a finite scheduling horizon that can be recursively rolled forward so that it addresses real-world operational needs. (ii) Another key objective is to introduce a proof of concept for a new methodology to integrate multi-stage stochastic programming with distributionally robust optimization to concurrently deal with different types of uncertainty. The proof of concept is given by a case study motivated by a partner surgical hospital which seeks to meet access delay targets and guarantee service within a safe target time window, in particular for acute patients.

4.1.1 Related Literature

Our work is related to multiple research areas, namely, appointment scheduling, healthcare coordination, distributionally robust optimization, and stochastic programming.

Appointment scheduling and healthcare coordination. There is a growing literature on appointment scheduling in healthcare operations with surveys provided by [44], [79], [123], and [7]. Some of the recent papers include [147], [109], [106], [54], [154], [128], [111], and [113].

[80] define two access delay types. *Direct access delay* is the time between the patient's arrival to the clinic on the day of her appointment and the time the doctor sees her; *indirect access delay* is the time between the patient's appointment referral and the time of her scheduled appointment. Most works have concentrated on direct waiting times and far fewer considered indirect waiting times. [137] propose a Markov decision process (MDP) to develop policies that minimize the number of patients that do not get a single appointment by a clinically determined maximum wait time target. [80] study an MDP under patients' preference for a clinic to decide how to manage access to its slots when patients can choose

between a single same-day or future appointment. [114] present an MDP under no-show and cancellation to allocate each patient a single appointment date within a specific horizon. [145] extend the work of [137] to require a sequence of appointment visits (with certain duration) for each patient while reducing access delays. They assume multiple identical therapy machines and are thus able to model total capacity by aggregating individual capacities of machines, unlike our paper. [151] consider a deterministic number of chemotherapy patients with multiple visits over time and formulate an optimization model to minimize access delay from their earliest start dates. [74] study a similar model to that of [137] and consider different patient types that require different levels of access to a single appointment. [60] develop an MDP under no-shows where patients undergo a series of assessments before being eligible for a surgery.

Our paper belongs to the stream of research on indirect access time. There are a number of key differences between the above papers and ours. First, we consider multiple non-identical surgeons as scarce resources as opposed to [145], which models either one single resource or multiple identical resources that are aggregated. Second, they assume that each patient needs either one (e.g., [137], [80], [114], and [74]) or multiple visits (e.g., [145], [151], and [60]), and these are all assumed to be known at the time of receiving the request. But, in our problem, each patient requires a clinic visit, which may or may not be followed by a surgery, and the need for surgery is realized at the clinic visit. Third, we do not consider no-shows and cancellations because they rarely occur in the settings of our highly specialized partner clinics. Fourth, an important goal of our study is to achieve timely access to care using *access delay targets* and model uncertainty in surgery duration, which are not considered in [60] and [145].

We address *healthcare coordination* in the sense of setting appointments for pairs of sequential visits that together achieve timely access to care. We found only two articles in this regard. [154] propose a coordinated pre-operative scheduling approach to evaluate patients' conditions prior to surgery. They model a two-station stochastic network, where each clinic may be staffed by multiple parallel providers and patients see the first available one. They give a myopic scheduling policy due to their complex setting. [98] use a simulation approach to evaluate their proposed heuristic policies for coordinating clinic and surgery visits. Our work develops optimization models rather than heuristics for determining the clinic and surgery visit decisions, and mathematically models uncertainty in both surgery duration and arrival process. This provides a general approach to a broader range of systems and parameters because heuristics do not readily extend to new settings.

Stochastic programming and distributionally robust optimization. As an alternative to MDP approaches and simulation to address uncertainty, two-stage stochastic

programming is usually employed to formulate appointment scheduling operational problems that incorporate uncertainty (see e.g., [55], [120], [25], and [135]). However, the uncertainty in stochastic parameters such as demand is often realized *progressively*, and the decision at each stage should be a function of the observed feedback outcomes up to that stage. Multi-stage stochastic programming (MSSP) is a more suitable approach for modeling such a setting (see e.g., [63]), which is the case in our paper.

Surgery durations across different patient classes and surgeons are not usually homogeneous; thus, it is challenging to characterize their exact probability distributions. To overcome this issue, distributionally robust optimization (DRO) approaches have recently received more attention. They optimize the worst-case performance over an ambiguity set, which represents a class of probability distributions with specified moment information. [102] formulate a DRO model in which an ambiguity set is used to include all distributions of service times with common mean and covariance and derive a semidefinite programming relaxation. [118] consider a similar problem except that service durations are independently distributed, and reformulate their DRO model as a conic program. However, their formulation requires the assumption that service durations could take on negative values. [91] consider a single-server DRO scheduling problem given a fixed sequence of appointments with ambiguous no-shows and service durations and derive mixed-integer nonlinear programs.

There are key differences between the above papers and ours. First, the focus of their DRO models is not on real-world settings; however, we develop a DRO formulation for an appointment scheduling problem with realistic features motivated by our partner hospital. Second, we develop a new approach which integrates a DRO model with an MSSP model to incorporate different types of uncertainty as well. Third, we leverage a set of transformations to turn our non-linear program into a tractable one, which can be efficiently solved by a new constraint generation algorithm.

The decisions made by most decision making under uncertainty approaches are often not implementable in practice. Rolling horizon type algorithms are usually developed to deal with this issue. For example, the rollout method for approximate dynamic programming ([27], [28] and [29]), Monte Carlo search tree method for reinforcement learning ([37], [71], and [129]) and rolling-horizon policies for MDPs ([86] and [10]) are three main applications of this idea. However, we extend the idea of rolling horizon into our IMSDRO approach as a way of adapting to the effect of uncertainty in the novel case of MSSP integrated with DRO.

4.1.2 Main Contributions and Focus

Below, we summarize the major contributions of this paper to the existing literature.

(1) Integrated multi-stage stochastic and distributionally robust optimization.

We believe that methodologically this paper is the first to develop an integrated multi-stage stochastic and distributionally robust optimization approach to simultaneously model two different types of uncertainties, namely the uncertainty in arrival process and the uncertainty in service time. While arrivals are often approximated by a Poisson process in operations models, in many services such as healthcare, the service time can depend greatly on what type of service is provided and by whom [79]. In the context of a specialized surgical unit, several types of surgeries may be offered by a number of surgeons, making it challenging to elicit a complete probability distribution of surgery duration for all surgery type-surgeon combinations. Hence, a DRO approach that only relies on limited distributional information (e.g., mean, standard deviation, and range) combined with an MSSP model to model the arrival process is extremely valuable. In this paper, we first develop an MSSP model that defines the decisions to be made at each stage as a function of the observed outcomes up to that stage and models the uncertainty around appointment request arrivals by a Poisson process from which we can take enough random samples to make a *scenario tree*. We integrate this MSSP model with a DRO approach, which makes no assumption on the exact probability distribution of surgery duration. Instead, it describes a *moment-based ambiguity set*, which captures a class of distributions with specified moment information. The exact formulation derived by the IMSDRO approach is not tractable. We leverage a set of transformations to turn this non-linearity into an approximate model that contains an embedded mixed-integer linear program in its constraints. We develop a new constraint generation algorithm that generates effective *scenario cuts* through this embedded optimization problem, to efficiently solve the model. Our IMSDRO methodology is flexible and can be applied to other service operations in which different types of uncertainty are to be modeled simultaneously. Our transformations can also be used for many other DRO models to turn them into tractable ones.

(2) Data-driven rolling horizon procedure. Since the decisions obtained by the IMSDRO approach are scenario dependent, they are not readily implementable in practice. We propose a *data-driven rolling horizon procedure* (RHP), which provides a framework to (i) make the decisions of the IMSDRO approach *implementable* in real practices, and (ii) empirically evaluate the performance of the scheduling policies obtained by the IMSDRO approach. The main advantage of the RHP is that it allows practitioners to make use of the latest information that is revealed as time unfolds and adjust their decisions by dynamically utilizing the realization of uncertain parameters. This RHP resolves the critical limitation of traditional stochastic programming policies, which are only valid for a limited number of scenarios. While the rolling horizon idea is, in general, similar to that of a rollout policy for constrained dynamic programs, a Monte Carlo search tree in reinforcement learning,

and rolling-horizon MDPs, implementation of a data-driven rolling horizon procedure in the context of IMSDRO is novel.

(3) Healthcare coordination for timely access delay. This paper presents a class of scheduling policies that aim to coordinate clinic consultation and surgical appointments in a specialized surgical setting to accommodate patients of different acuity/priority levels within a predefined priority-based time window. The need to consider *care coordination* has been raised by [123] and emphasized by the recent survey of [7]. To the best of our knowledge, this paper is the first work to date that uses optimization approaches to study and model the impact of clinic and surgery appointment coordination to accommodate priority-based access delay targets. In §4.6, we demonstrate that coordinating clinic and surgery appointments in a specialized surgical unit using our new IMSDRO methodology can significantly improve surgical access delay for patients with acute conditions. We show that there is a trade-off between meeting access delay targets and incurring overtime. This allows decision makers to define a set of access delay targets that results in acceptable surgeon overtime. Although our work is motivated by healthcare, our models, methodology, and insights can also be extended to the general appointment-based service systems. We discuss several important practical implications and insights from our work in §4.7.

4.2 Problem Statement

In this section, we present the description and specifications of the CAS problem. This new problem is motivated by a real-world healthcare scheduling application in our collaborating hospitals.

Surgeons offer both clinic consultation and surgical procedures. There are a number of surgeons who work in two clinic and surgery teams. The set of surgeons is denoted by $\mathcal{K}$ where each surgeon is presented by $k \in \mathcal{K}$. Each period is typically a day. We use the terms day and period interchangeably. On any given day, one team sees patients in the clinic while the other team performs surgery in the operating rooms (ORs). Each surgeon switches between clinic and surgery teams on the following day and maintains his/her own clinical and surgical calendars. This is called an *every-other-day operating calendar* for surgeons. The system allows both clinical and surgical overtimes along with the regular clinical and surgical capacities for surgeons. Each surgeon $k \in \mathcal{K}$ has a regular clinical capacity of U_m^k on clinic day m, and a regular surgical capacity of V_n^k on surgery day n. These details can be easily modified to accommodate other healthcare settings.

Patients from different classes/types. There are different classes of patients whose requests are received by the surgical clinic. The set of *patient classes* is denoted by Γ where

each patient class is presented by a tuple $\gamma = (\phi, \nu) \in \Gamma$, where ϕ is the *referral type* (i.e., local or remote) and ν is the *indications of disease* (e.g., colon cancer, rectal prolapse, ulcerative colitis, diverticulitis, etc). Patients are referred either by other hospital units or by their primary care physicians to the surgical clinic to consult with a surgeon and evaluate the need for a surgery. When the surgical clinic receives an appointment request, the patient's electronic health records reveal the indication of disease as well as whether the patient is locally or remotely referred. The referral type and the indication of diseases together determine a patient's class. We use the terms request and referral interchangeably. Each appointment request first requires a clinic consultation appointment with a surgeon, which then may need to be followed by a surgery with a probability r_γ for each class γ. A patient class determines the probability r_γ that a patient will need a surgery. The surgery probability helps approximate the required surgery workload in the future. The decision of whether a patient requires a surgery is made during the patient's clinic visit. If a surgery is required, we assume that it has to be performed by the same surgeon who visited the patient at the clinic visit. This feature captures the *continuity of care* between patient-surgeon and is often preferred by patients since the patient has already established some trust and a relationship with the surgeon.

Various types of uncertainty. In light of the availability of historical data and the inherent uncertainty of the system, there are three types of uncertainty. The first is the total number of appointment requests received from each patient class in each period, which is realized at the end of the period. The second is whether a given patient requires a surgery or not, which is revealed at the clinic visit. If a surgery is needed, the third type of uncertainty concerns the surgery duration.

We know the probability distribution for the number of appointment requests made by each patient class in each period from which a set $\mathcal{S}$ of *stochastic scenarios* (indeed a scenario tree) is generated to model the existing uncertainty in appointment request arrivals. We represent such uncertainty by $\mathcal{D}^s_{\gamma,t}$, which is the set of class $\gamma \in \Gamma$ patients whose request is received on any day t under scenario $s \in \mathcal{S}$ (see §4.3.1). Nonetheless, we have limited distributional information on the distribution of surgery duration $d_{\gamma,k}$ for each patient class $\gamma \in \Gamma$ and surgeon $k \in \mathcal{K}$ pair. There is usually a wide range of patient classes served by several different surgeons, which leads to having only a limited number of cases/examples for each patient class-surgeon combination. This makes it hard to fit distributions tailored to each surgeon and patient class pair, because individual surgeons may perform many surgery types with small annual volumes (see discussion in §4.1). A *moment-based ambiguity set* is employed to incorporate all such distributions with a common mean, standard deviation, and support (see §4.3.2). We employ a multi-stage decision-making setting as well because

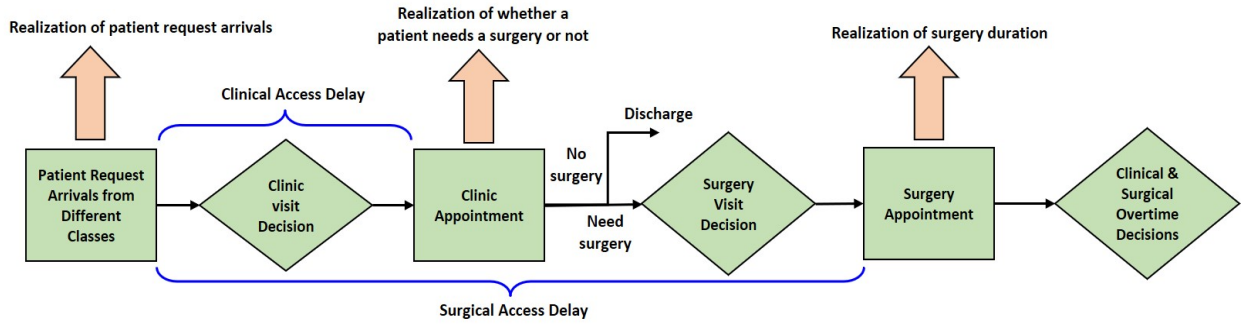


Figure 4.1: The illustration of sequence of events, timing of different uncertainty realizations and proactive clinic and surgery scheduling decisions made for each patient request in the surgical clinic.

the uncertainty in parameters is progressively realized in each period. We then develop an integrated optimization-based approach, denoted above as the IMSDRO approach, combining an MSSP model with a DRO approach to simultaneously model all types of uncertainty. The goal is to find an optimal clinic and (if needed) surgery visit date for each patient with minimum overtimes for surgeons such that class-specific access delay targets are met for patients. We denote the clinical and surgical overtimes of surgeon $k \in \mathcal{K}$ on clinic day m and surgery day n under scenario $s \in \mathcal{S}$ by $q^k_{m,s}$ and $o^k_{n,s}$, respectively.

Clinical and surgical decisions. Figure 4.1 illustrates the sequence of events and decisions, surgical and clinical access delays and timing of uncertainty realizations in the CAS problem. At the end of each period, the uncertainty about the number of appointment requests/referrals received on that period is realized and the clinic appointments for those patients are scheduled at the end of that period. The clinic visit day is promised at the end of the arrival day and is denoted by $x^{k,m}_{p,\gamma,t,s}$, which is whether a class γ patient $p \in \mathcal{D}^s_{\gamma,t}$ whose request was received on any day t under scenario s, has clinic visit on day m with surgeon k. The next decision, the day of surgery, is made on the clinic appointment day. After completing the clinic visit, it becomes known whether the patient requires a surgery. If the patient needs a surgery, we schedule a surgery appointment, which must be with the same surgeon with whom she/he had the clinic visit. The surgery decision is denoted by $y^{k,n}_{\gamma,t,m,s}$, which is the number of class γ patients whose requests were received on any day t under scenario s and had clinic visit on m, and we choose surgery day n with surgeon k. After the realization of surgery need and duration, we calculate two auxiliary variables of the clinical and the surgical overtimes $q^k_{m,s}$ and $o^k_{n,s}$ of surgeons, respectively.

Timely access to care. To ensure that patients are granted timely access to care, we place hard constraints on the allowable time intervals during which a patient may have clinic and surgery visits safely. For each patient class γ, we define a parameter called WTC_γ or

"*minimum wait time target for the clinic visit of a patient class γ patient.*" For example, in our case study (see §4.6), the value of this parameter only depends on ϕ as it was appropriate to assume the wait time to clinic is determined based on whether the patient is referred locally or remotely to our partner hospital's hospital. In particular, for the local referral, WTC_γ is zero because the patient is physically at or around the surgical clinic. But, for the remote referral, we allow a minimum of WTC_γ days (5 days in the case study) from when a patient referral is received until her/his clinic visit so as to give the patient time to make travel arrangement to the surgical clinic. We also define another important parameter called WTS_γ or "*maximum wait time target to surgery visit.*" This can be thought of as the maximum wait time that the patient's surgery, if needed, can be safely postponed from the time of patient referral. Our methodology ensures that all patients are offered at least one surgery visit within their WTS_γ. Based on the recommendation of our partner clinic, we define a parameter CSG_γ or "*minimum gap between clinic and surgery visits of a class γ patient.*" This corresponds to the minimum required number of days between the patient's clinic and surgery visits. While this can be zero, some surgeries require a period of preparation prior to the surgery. WTC_γ, WTS_γ and CSG_γ are set by the surgical clinic in our case study, but can be easily modified in other settings.

According to the above-defined parameters, if we receive a referral on any period t from patient class γ, we define (i) the earliest time $EC_{\gamma,t} = t + WTC_\gamma$, and the latest time $LC_{\gamma,t} = t + WTS_\gamma - CSG_\gamma$ for setting the clinic appointment, and (ii) the earliest time $ES_{\gamma,t} = t + WTC_\gamma + CSG_\gamma$ and the latest time $LS_{\gamma,t} = t + WTS_\gamma$ for choosing the surgery appointment (if needed) for this patient. In our approach, both clinic and surgery appointments are scheduled within these *clinical and surgical target time windows* that depend on the appointment request period. This requirement on each patient's flow pathway adds significant complexity to the CAS problem. Figure 4.2 depicts the allowable target time windows for having the clinic and surgery (if needed) appointments for a typical patient from class γ whose appointment request is received on any period t.

4.3 Integrated Multi-stage Stochastic and Distributionally Robust Optimization Methodology

Analytics Overview. In this section, the IMSDRO methodology for the CAS problem described in §4.2 is presented. In §4.3.1, we first assume that the surgery duration is deterministic and develop a Multi-stage Stochastic Mixed-Integer Program (MS-MIP) in which a scenario tree is exploited to model the uncertainty in the number of patient appointment requests on each period. In §4.3.2, we then extend this MS-MIP model to account for un-

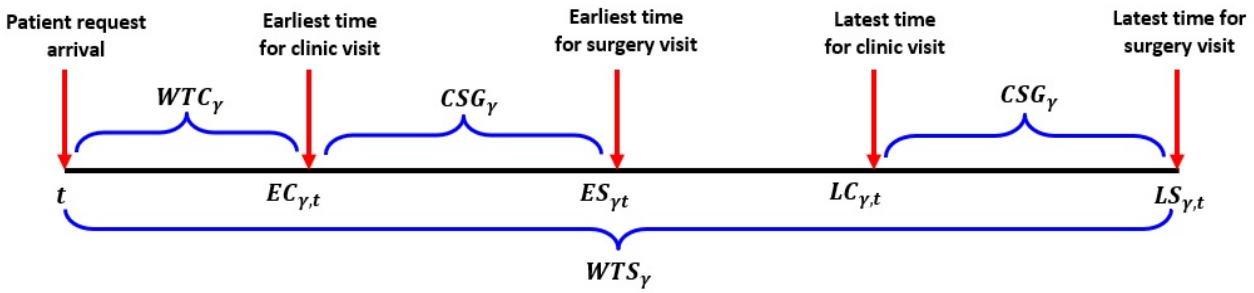


Figure 4.2: The illustration of minimum Wait Time for Clinic visit (WTC), minimum Clinic to Surgery visits Gap (CSG), maximum Wait Time to Surgery visit (WTS) for a patient whose request is received on any period t.

certainty in the surgery duration by developing a DRO approach that uses an ambiguity set constructed based on the empirical mean, standard deviation, and support of the surgery duration. Given that the resulting formulation is not tractable, we deploy a set of approximations based on the structural properties and a scenario cut-generating model, which results in an approximate tractable reformulation (IMSDRO-APRX).

4.3.1 Multi-stage Stochastic Mixed-Integer Program Model

We define three horizons (see Figure 3 for details) for the CAS problem: (i) current scheduling horizon $\mathcal{L}$, (ii) current arrival horizon $\mathcal{T}$, and (iii) past arrival horizon $\mathcal{U}$. Through using this modeling approach, we account for initial steady-state clinical and surgical workloads. The *current scheduling horizon* $\mathcal{L}$ is the set of periods from current period t_0 until period t_e, over which we decide the clinic and surgery appointment dates for patients. There are two types of *patient arrival horizons*: (i) the *"current" arrival horizon* $\mathcal{T}$ is the set of periods from current period t_0 until period t_b for new patient request arrivals, which is the first portion of the current scheduling horizon, and (ii) the *"past" arrival horizon* $\mathcal{U}$ is the set of periods from period 1 until period $t_0 - 1$ for past patient request arrivals over the previous scheduling horizon. The reason for defining the set $\mathcal{U}$ is twofold. First, the clinic visit of a patient whose request has been already received in $\mathcal{U}$ may happen on any period (day) in $\mathcal{T}$, so we may still need to make a surgery visit decision. Second, the surgery visit of the patients whose request is received in $\mathcal{U}$ may happen on any period in $\mathcal{T}$ and they consumes surgery capacity during the horizon $\mathcal{T}$. Because of the access wait time targets to surgery, $|\mathcal{L}| = |\mathcal{T}| + \max_\gamma\{WTS_\gamma\}$ is the required length of current scheduling horizon. Note that t_0 and t_b ($t_b = t_0 + |\mathcal{T}| - 1$) are specified by the surgical clinic to determine the first and last periods for new patient request arrivals, and $t_e = t_0 + |\mathcal{T}| + \max_\gamma\{WTS_\gamma\}$.

A multi-stage stochastic program allows us to have several decision layers, where random outcomes are *progressively* realized, and the clinical and surgical decisions should be adapted

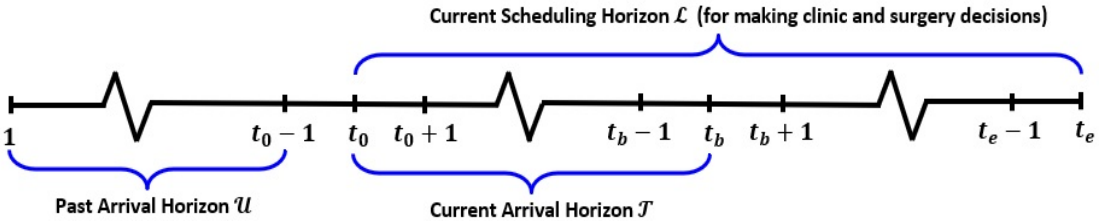


Figure 4.3: The illustration of arrival horizon $\mathcal{U}$ for patient request arrivals in the previous scheduling horizon, arrival horizon $\mathcal{T}$ for patient request arrivals in the current scheduling horizon, and current scheduling horizon $\mathcal{L}$.

to this process. In general, a T-stage stochastic program includes a sequence of stochastic parameters $\xi_1, \xi_2, \cdots, \xi_{T-1}$ with a discrete support. A *scenario* is a realization of these stochastic parameters, and a *scenario tree* represents the progressive observation of random parameters. To model stochasticity in the number of appointment request arrivals as a scenario tree, a set of scenarios $\mathcal{S}$ with a countable size $S = |\mathcal{S}|$ is defined. The corresponding scenarios' probabilities are $\pi_1, \pi_2, \cdots, \pi_S$, and a realization of the stochastic parameters for scenario $s \in \mathcal{S}$ is presented by $(\xi^s_{t_0}, \xi^s_{t_0+1}, \cdots, \xi^s_{t_b})$ where $\xi^s_t = (\mathcal{D}^s_{\gamma,t} : \gamma \in \Gamma)$ is a realization for the number of requests on period $t \in \mathcal{T}$ over different classes under scenario $s \in \mathcal{S}$, and $\mathcal{D}^s_{\gamma,t}$ is the stochastic set of class γ patients whose service request/referral is received in period $t \in \mathcal{T}$ under scenario $s \in \mathcal{S}$. Noting that $\xi^s_{t_0}$ is the same (deterministic) for all scenarios $s \in \mathcal{S}$ because it is the number of appointment requests in the current period t_0 of the arrival horizon $\mathcal{T}$. Moreover, we define $\widetilde{\mathcal{D}}_{\gamma,t}$ as the deterministic set of class $\gamma \in \Gamma$ patients whose request was already received on day $t \in \mathcal{U}$. Formally, we have the following stochasticity assumption for the number of appointment requests in the CAS problem.

Assumption 1 (Stochasticity Assumption). There is full distributional information for the number of patient appointment requests in every period over the current arrival horizon $\mathcal{T}$. Such uncertainty is modeled by a stochastic process ξ with a realization of stochastic parameters presented by $(\xi^s_{t_0}, \xi^s_{t_0+1}, \cdots, \xi^s_{t_b})$ with a probability π_s under scenario $s \in \mathcal{S}$, where $\xi^s_t = (\mathcal{D}^s_{\gamma,t} : \gamma \in \Gamma)$ is a realization for the number of patient appointment requests on period $t \in \mathcal{T}$ under scenario $s \in \mathcal{S}$.

In an MSSP, a policy should be *non-anticipative*, meaning that the decisions made at each stage must not be dependent on the future realization of stochastic parameters. There are two common ways for formulating an MSSP [61]. In the first, an MSSP is formulated as a sequence of nested two-stage stochastic programs in which non-anticipativity is implicitly imposed. In the second (used in this paper), a set of *non-anticipativity constraints* (NAC) is explicitly modeled.

Figure 4.4 (left-hand side) shows an example of a scenario tree with four stages and

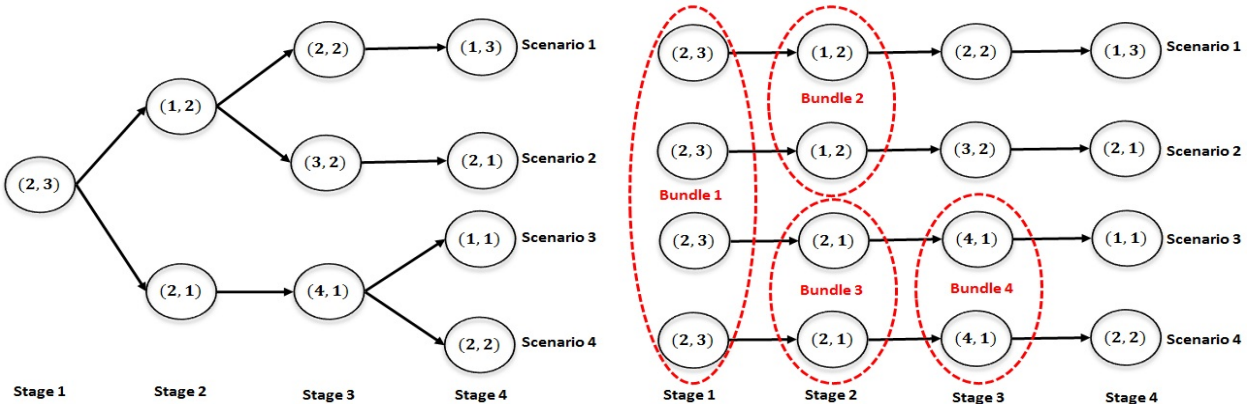


Figure 4.4: (LHS): An illustration of a scenario tree for the number of appointment request arrivals of 2 patient classes in a 4-stage MSSP with 4 scenarios where in each node (i, j) shows the number of appointment request arrivals of patient classes 1 and 2 at each stage t and scenario s, and (RHS): the corresponding scenario fan with four scenario bundles required for this 4-stage MSSP. The dashed ovals covering the nodes present NACs.

four scenarios for the CAS problem with two classes. In each scenario node, there is a realization $(|\mathcal{D}^s_{1,t}|, |\mathcal{D}^s_{2,t}|)$ where $|\mathcal{D}^s_{1,t}|$ and $|\mathcal{D}^s_{2,t}|$ are the number of class 1 and 2 appointment requests that are received on period $t \in \mathcal{T}$, respectively. For example, $\mathcal{D}^2_{1,3} = \{1, 2, 3\}$ and $\mathcal{D}^2_{2,3} = \{1, 2\}$ are for the node at stage three and scenario two. Figure 4.4 (right-hand side) is an alternative representation of the scenario tree, which is called *scenario fan*, where the individual scenarios observed in the particular stages are aggregated over all periods to form four scenarios. However, this scenario fan is *not permissible*. If we solve the CAS problem for each of the scenarios, the solution found might not be feasible for the overall problem because they imply decisions that anticipate future uncertain events. Thus, we need to enforce NACs to have permissible decisions. The dashed ovals covering the nodes represent NACs. For example, since all four scenarios have the same realizations at stage 1, they share the same scenario bundle, and so a NAC is imposed to guarantee that the same surgical and clinical decisions are made at all nodes in this scenario bundle. This is the same for scenarios 1 and 2 on period $t = 1$, scenarios 3 and 4 on period $t = 2$, and scenarios 3 and 4 on period $t = 3$.

The other notations are given in Table 4.1. Tilda ($\sim$) is used to distinguish *decisions* x and y from *parameters* $\tilde{x}$, $\tilde{y}$, and $\tilde{z}$. In decisions x and y, the *subscript* indices are the given information, and *superscript* indices show the decisions. For example, in $x^{k,m}_{p,\gamma,t,s}$, the given information is the type γ patient p whose request is received on period t under scenario s, and then we make the decision about clinic date m and surgeon k. Also, decision $\widehat{y}$ is similar to decision y except it applies only to arrivals on current period t_0, which is deterministic (hence no dependence on scenario). We use bold notations whenever some indices of parameters/variables are removed.

The proposed MS-MIP model for the CAS problem is presented as follows, noting that the surgery duration $d_{\gamma,k}$ is assumed to be deterministic in this formulation.

Indices	
t, m, n	: Day indices (t is used for the day that an appointment request is received, and m and n are used for a clinic day and a surgery day, respectively.)
γ	: Patient class index, $\gamma = (\phi, \nu) \in \Gamma$ (ϕ is the referral type and ν is the disease indications).
k	: Surgeon index, $k \in \mathcal{K}$.
p	: Patient index, $p \in \mathcal{D}^s_{\gamma,t}$.
s	: Scenario index, $s \in \mathcal{S}$.
Deterministic and Stochastic Parameters	
U^k_m	: Total clinical capacity of surgeon $k \in \mathcal{K}$ on clinic day $m \in \mathcal{L}$.
V^k_n	: Total surgical capacity of surgeon $k \in \mathcal{K}$ on surgery day $n \in \mathcal{L}$.
c_γ	: Clinic duration of a patient class $\gamma \in \Gamma$.
$d_{\gamma,k}$	: Surgery duration of a patient class $\gamma \in \Gamma$ performed by surgeon $k \in \mathcal{K}$.
$\mathcal{D}^s_{\gamma,t}$	: Set of class $\gamma \in \Gamma$ patients whose request is received on day $t \in \mathcal{T}$ under scenario $s \in \mathcal{S}$.
$\widetilde{\mathcal{D}}_{\gamma,t}$	: Set of class $\gamma \in \Gamma$ patients whose request is already received on day $t \in \mathcal{U}$.
π_s	: Probability of occurrence of scenario $s \in \mathcal{S}$.
r_γ	: Surgery probability of a class $\gamma \in \Gamma$ patient.
$\widetilde{x}^{k,m}_{p,\gamma,t}$	: Binary parameter equal to 1 if a class $\gamma \in \Gamma$ patient p whose request was received on day $t \in \mathcal{U}$ has clinic visit on day $m \in \mathcal{T}\backslash\{t_0\}$ with surgeon $k \in \mathcal{K}$, and zero otherwise.
$\widetilde{z}^k_{\gamma,t}$	: The number of class $\gamma \in \Gamma$ patients whose request is received on day $t \in \mathcal{U} \cup \{t_0\}$, and has clinic visit on day t_0 with surgeon $k \in \mathcal{K}$, and also needs surgery.
$\widetilde{y}^{k,n}_{\gamma,t,m}$	: The number of class $\gamma \in \Gamma$ patients whose request is received on day $t \in \mathcal{U}$, and has clinic visit on day $m \in \mathcal{U}$, and surgery visit on day $n \in \mathcal{T}$ with surgeon $k \in \mathcal{K}$.
Stage Decision Variables	
$x^{k,m}_{p,\gamma,t,s}$	: Binary variable equal to 1 if a class $\gamma \in \Gamma$ patient p whose request is received on day $t \in \mathcal{T}$ under scenario $s \in \mathcal{S}$ has clinic visit on day $m \in \mathcal{L}$ with surgeon $k \in \mathcal{K}$, and 0 otherwise.
$y^{k,n}_{\gamma,t,m,s}$	: The number of class $\gamma \in \Gamma$ patients whose requests are received on day $t \in \mathcal{U} \cup \mathcal{T}$ under $s \in \mathcal{S}$, and have clinic visit on $m \in \mathcal{T}\backslash\{t_0\}$, and surgery visit on $n \in \mathcal{L}$ with surgeon $k \in \mathcal{K}$.
$\widehat{y}^{k,n}_{\gamma,t,t_0}$	: The number of class $\gamma \in \Gamma$ patients whose requests are received on day $t \in \mathcal{U} \cup \{t_0\}$, and have clinic visit on day t_0, and surgery visit on $n \in \mathcal{L}$ with surgeon $k \in \mathcal{K}$.
$q^k_{m,s}$	: Clinical overtime of surgeon $k \in \mathcal{K}$ on the clinic day $m \in \mathcal{L}$ under $s \in \mathcal{S}$.
$o^k_{n,s}$	: Surgical overtime of surgeon $k \in \mathcal{K}$ on the surgery day $n \in \mathcal{L}$ under $s \in \mathcal{S}$.

Table 4.1: The description of indices, parameters and decisions of the MS-MIP model for the CAS problem.

Objective function. The objective (4.1) of the MS-MIP model is to minimize the expected total clinical and surgical overtimes of all surgeons over the scheduling horizon and scenarios.

$$\min \quad \sum_{s\in\mathcal{S}} \sum_{k\in\mathcal{K}} \pi_s \Big(\sum_{m\in\mathcal{L}} q^k_{m,s} + \sum_{n\in\mathcal{L}} o^k_{n,s} \Big). \tag{4.1}$$

Constraints for access delay to clinic appointments. Constraints (4.4)-(4.5) below guarantee that the clinic visit decision or $x^{k,m}_{p,\gamma,t,s}$ for each class γ patient whose request is received on any day t under scenario s should be an available date between the earliest clinic time $EC_{\gamma,t} = t + WTC_\gamma$ and the latest clinic time $LC_{\gamma,t} = t + WTS_\gamma - CSG_\gamma$ (see Figure

4.2 for the feasible clinic range).

$$x^{k,m}_{p,\gamma,t,s} = 0, \quad \forall \gamma \in \Gamma, t \in \mathcal{T}, s \in \mathcal{S}, p \in \mathcal{D}^s_{\gamma,t}, k \in \mathcal{K}, \\ m \in \big[t_0, t + WTC_\gamma - 1\big] \cup \big[t + WTS_\gamma - CSG_\gamma + 1, t_e\big]. \tag{4.2}$$

$$\sum_{m=t+WTC_\gamma}^{t+WTS_\gamma-CSG_\gamma} \sum_{k\in\mathcal{K}} x^{k,m}_{p,\gamma,t,s} = 1, \ \ \forall \gamma \in \Gamma, t \in \mathcal{T}, s \in \mathcal{S}, p \in \mathcal{D}^s_{\gamma,t}. \tag{4.3}$$

$$x^{k,m}_{p,\gamma,t,s} = 0, \ \ \forall \gamma \in \Gamma, t \in \mathcal{T}, s \in \mathcal{S}, p \in \mathcal{D}^s_{\gamma,t}, k \in \mathcal{K}, m = t_0, \cdots, t + WTC_\gamma - 1. \tag{4.4}$$

$$\sum_{m=t+WTC_\gamma}^{t+WTS_\gamma-CSG_\gamma} \sum_{k\in\mathcal{K}} x^{k,m}_{p,\gamma,t,s} = 1, \ \ \forall \gamma \in \Gamma, t \in \mathcal{T}, s \in \mathcal{S}, p \in \mathcal{D}^s_{\gamma,t}. \tag{4.5}$$

$$x^{k,m}_{p,\gamma,t,s} = 0, \ \ \forall \gamma \in \Gamma, t \in \mathcal{T}, s \in \mathcal{S}, p \in \mathcal{D}^s_{\gamma,t}, k \in \mathcal{K}, m = t + WTS_\gamma - CSG_\gamma + 1, \cdots, t_e. \tag{4.6}$$

Constraints for access delay to surgery appointments. Constraints (4.7)-(4.9) state that the surgery of each patient is performed by the same surgeon who performed the associated clinic visit, and the surgery visit for each patient should be within a clinically safe range of days (see Figure 4.2 for the feasible surgery range). More explicitly, the class γ patients whose requests are received on either day $t \in \mathcal{U}$ or day $t \in \mathcal{T}$, and have clinic visit on day $m \in \mathcal{T}$ could have their surgery visit on any day between $m + CSG_\gamma$ and $t + WTS_\gamma$. We denote the surgery visit decisions by $y^{k,n}_{\gamma,t,m,s}$ and $\widehat{y}^{k,n}_{\gamma,t,t_0}$ for patients whose clinic visit is any day $m \in \mathcal{T}\backslash\{t_0\}$ and the current day t_0, respectively.

$$r_\gamma\Big(\sum_{p\in\widetilde{\mathcal{D}}_{\gamma,t}} \widetilde{x}^{k,m}_{p,\gamma,t}\Big) \ \le \sum_{n=m+CSG_\gamma}^{t+WTS_\gamma} y^{k,n}_{\gamma,t,m,s}, \ \ \forall \gamma \in \Gamma, t \in \mathcal{U}, m \in \mathcal{T}\backslash\{t_0\}, k \in \mathcal{K}, s \in \mathcal{S}. \tag{4.7}$$

$$r_\gamma\Big(\sum_{p\in\mathcal{D}^s_{\gamma,t}} x^{k,m}_{p,\gamma,t,s}\Big) \ \le \sum_{n=m+CSG_\gamma}^{t+WTS_\gamma} y^{k,n}_{\gamma,t,m,s}, \ \ \forall \gamma \in \Gamma, t \in \mathcal{T}, m \in \mathcal{T}\backslash\{t_0\}, k \in \mathcal{K}, s \in \mathcal{S}. \tag{4.8}$$

$$\widetilde{z}^{k}_{\gamma,t} \ \le \sum_{n=t_0+CSG_\gamma}^{t+WTS_\gamma} \widehat{y}^{k,n}_{\gamma,t,t_0}, \ \ \forall \gamma \in \Gamma, t \in \mathcal{U} \cup \{t_0\}, k \in \mathcal{K}. \tag{4.9}$$

Note that the constraints (4.7) are for the class γ patients $\widetilde{\mathcal{D}}_{\gamma,t}$ whose request is received in the *previous* arrival horizon $\mathcal{U}$ (so they are already in the system) and their clinic visits are denoted by parameter $\widetilde{x}^{k,m}_{p,\gamma,t}$, and their surgery is being made on one day in the current horizon $\mathcal{T}$. However, the constraints (4.8) are for the class γ patients $\mathcal{D}^s_{\gamma,t}$ whose request is

received in the *current* arrival horizon $\mathcal{T}$ under scenario s and their clinic visits are denoted by $x^{k,m}_{p,\gamma,t,s}$. In both constraints (4.7) and (4.8), the clinic appointment of patients may happen on any day over horizon $\mathcal{T}\backslash\{t_0\}$, so their surgery need is specified by a surgery probability r_γ as their clinic visit has not happened yet. The constraints (4.9) are for the patients denoted by parameter $\widetilde{z}^k_{\gamma,t}$ whose request is received on any day in horizon $\mathcal{U}\cup\{t_0\}$ (so they are already in the system), but unlike the constraints (4.7)-(4.8), their clinic visit is on the current day t_0, and hence their surgery need is realized.

Clinical and surgical capacity constraints. Constraints (4.10)-(4.11) below restrict the amount of clinical and surgical workloads (both regular capacity and overtime) for each surgeon $k \in \mathcal{K}$ on each day $n \in \mathcal{L}$, respectively, over the scheduling horizon for the CAS problem.

$$\sum_{\gamma\in\Gamma} c_\gamma\Big(\sum_{t\in\mathcal{U}}\sum_{p\in\widetilde{\mathcal{D}}_{\gamma,t}} \widetilde{x}^{k,m}_{p,\gamma,t} + \sum_{t\in\mathcal{T}}\sum_{p\in\mathcal{D}^s_{\gamma,t}} x^{k,m}_{p,\gamma,t,s}\Big) \;\le\; U^k_m + q^k_{m,s},\ \forall m\in\mathcal{L}, k\in\mathcal{K}, s\in\mathcal{S}. \tag{4.10}$$

$$\sum_{\gamma\in\Gamma} d_{\gamma,k}\Big(\sum_{t\in\mathcal{U}}\sum_{m\in\mathcal{U}} \widetilde{y}^{k,n}_{\gamma,t,m} + \sum_{t\in\mathcal{T}}\sum_{m\in\mathcal{T}\backslash\{t_0\}} y^{k,n}_{\gamma,t,m,s} + \sum_{t\in\mathcal{T}\cup\{t_0\}} \widehat{y}^{k,n}_{\gamma,t,t_0}\Big) \le V^k_n + o^k_{n,s}, \tag{4.11}$$

$\forall n\in\mathcal{L}, k\in\mathcal{K}, s\in\mathcal{S}.$

Non-anticipativity constraints. In any given stage over the scheduling horizon, the decision maker cannot foresee the future outcomes of the total number of appointment requests; therefore, the clinic and surgery decisions must satisfy NACs. This indicates that these decisions in a given stage t are identical for each pair (s, s') of scenarios with a common ancestor node in that stage (see Figure 4.4). If two scenarios s and s' share the same history of random parameters ξ^s and $\xi^{s'}$ up to stage t, then the decisions made at stage t are the same among all scenarios placed in the same scenario bundle. Constraints (4.12)-(4.13) are the corresponding NAC for the CAS problem.

$$x^{k,m}_{p,\gamma,t,s} = x^{k,m}_{p,\gamma,t,s'}, \forall k\in\mathcal{K}, \gamma\in\Gamma, m\in\mathcal{L}, t\in\mathcal{T}, p\in\mathcal{D}^s_{\gamma,t}, \tag{4.12}$$
$$s, s'\in\mathcal{S}, (\xi^s_{t_0+1},\cdots,\xi^s_t) = (\xi^{s'}_{t_0+1},\cdots,\xi^{s'}_t).$$
$$y^{k,n}_{\gamma,t,m,s} = y^{k,n}_{\gamma,t,m,s'}, \forall k\in\mathcal{K}, \gamma\in\Gamma, m\in\mathcal{T}\backslash\{t_0\}, t\in\mathcal{T}\cup\mathcal{U}, n\in\mathcal{L}, \tag{4.13}$$
$$s, s'\in\mathcal{S}, (\xi^s_{t_0+1},\cdots,\xi^s_t) = (\xi^{s'}_{t_0+1},\cdots,\xi^{s'}_t).$$

Note that we do not require defining the NACs for the other variables $q^k_{m,s}$ and $o^k_{n,s}$. The reason is because these auxiliary decisions are calculated directly from decisions $x^{k,m}_{p,\gamma,t,s}$ and $y^{k,n}_{\gamma,t,m,s}$ by the constraints (4.10)-(4.11), and thereby preserving the non-anticipativity for them automatically.

Other constraints. Constraints (4.14)-(4.17) define the binary and non-negativity restrictions on the clinic and surgery appointment decisions, and clinical and surgical overtimes, respectively.

$$x^{k,m}_{p,\gamma,t,s} \in \{0,1\},\ \forall k \in \mathcal{K}, m \in \mathcal{L}, \gamma \in \Gamma, t \in \mathcal{T}, p \in \mathcal{D}^s_{\gamma,t}, s \in \mathcal{S}. \tag{4.14}$$

$$y^{k,n}_{\gamma,t,m,s} \geq 0,\ \ \forall k \in \mathcal{K}, m \in \mathcal{T}\backslash\{t_0\}, \gamma \in \Gamma, t \in \mathcal{U} \cup \mathcal{T}, n \in \mathcal{L}, s \in \mathcal{S}. \tag{4.15}$$

$$\widehat{y}^{k,n}_{\gamma,t,t_0} \geq 0,\ \ \forall k \in \mathcal{K}, n \in \mathcal{L}, \gamma \in \Gamma, t \in \mathcal{U} \cup \{t_0\}. \tag{4.16}$$

$$q^k_{m,s}, o^k_{n,s} \geq 0,\ \forall k \in \mathcal{K}, m, n \in \mathcal{L}, s \in \mathcal{S}. \tag{4.17}$$

Remark (Patient-centered Care). It is worth noting that the proposed MS-MIP model (4.1)-(4.17) has been developed to be *patient-centered* by putting *hard constraints* on access delay targets, thus guaranteeing full service (i.e., clinic and surgery appointments) within a predefined priority-based safe interval (see Figure 4.2). It is, however, inevitable to employ clinical and surgical overtime on some days in order to attain this goal, and the best scheduling policy is thus the one that achieves such service level with the minimum possible clinical and surgical overtime. In general, there is a trade-off between access delay targets and surgeon overtime. The tighter the access targets, the higher the overtime. In Appendix C, we develop an alternative formulation for the CAS problem, that strikes a balance between meeting access delay targets and incurring clinical and surgical overtime. This alternative model allows decision makers to set penalties on violating access targets and incurring surgeon overtime.

4.3.2 Integrated Multi-stage Stochastic and Distributionally Robust Model

In this section, we extend the MS-MIP model (4.1)-(4.17), by incorporating ambiguous distributional information for surgery duration of each patient class and surgeon pair. Surgery duration is usually highly variable (see discussions in §4.1); however, there is often little uncertainty in clinic duration (e.g., in our partner hospitals, the clinic visits are scheduled in 15-minute time slots). The uncertainty in surgery duration is modeled by using an ambiguity set that is constructed based on the empirical mean, standard deviation, and support of the surgery duration. More precisely, besides the Stochasticity Assumption 1, another important assumption in the IMSDRO is as follows.

Assumption 2 (Ambiguity Assumption). There is ambiguous distributional information about the surgery duration of each patient class and surgeon pair. The variation in the surgery duration within each patient class and surgeon pair is insignificant. This limited distributional information includes two stochastic moments (i.e., mean and standard deviation), and the

support. A moment-based ambiguity set is used to model such uncertainty in the surgery duration.

From the Assumption 2, the surgery duration vector $\boldsymbol{d} = (d_{\gamma,k} : \gamma \in \Gamma, k \in \mathcal{K})$ for different classes and surgeons has an unknown probability distribution P with a *polyhedral support set* Θ as follows:

$$\Theta \;=\; \Big\{\boldsymbol{d} \in \mathbb{R}_{+}^{|\Gamma|\times|\mathcal{K}|} : d_{\gamma,k}^{LB} \;\leq\; d_{\gamma,k} \;\leq\; d_{\gamma,k}^{UB},\; \forall \gamma \in \Gamma, k \in \mathcal{K}\Big\}, \tag{4.18}$$

where $\boldsymbol{d}^{LB}$ and $\boldsymbol{d}^{UB} \in \mathbb{R}_{+}^{|\Gamma|\times|\mathcal{K}|}$ denote the lower and upper bound vectors for the surgery duration $\boldsymbol{d}$, respectively. Such lower and upper bounds can be computed from available historical data.

Definition 8 (**Marginal Moment-based Ambiguity Set**). *Given a set of $|L|$ observations of surgery duration $\boldsymbol{d}$, denoted by $\{\boldsymbol{d}^l\}_{l\in L}$ where $\boldsymbol{d}^l \in \mathbb{R}_{+}^{|\Gamma|\times|\mathcal{K}|}$, a moment-based ambiguity set $\Phi(\boldsymbol{\mu}, \boldsymbol{\sigma}, \Theta)$ is defined for the probability distribution P using the marginal mean vector $\boldsymbol{\mu} \in \mathbb{R}_{+}^{|\Gamma|\times|\mathcal{K}|}$ and standard deviation vector $\boldsymbol{\sigma} \in \mathbb{R}^{|\Gamma|\times|\mathcal{K}|}$ of these realizations of surgery durations as follows:*

$$\begin{aligned}
\Phi(\boldsymbol{\mu}, \boldsymbol{\sigma}, \Theta) \;:=\; & \Big\{P : \\
\int_{\Theta} dP(d) \;&=\; 1, && \text{(4.19a)}\\
\int_{\Theta} d_{\gamma,k}\; dP(d) \;&=\; \mu_{\gamma,k},\;\; \forall \gamma \in \Gamma, k \in \mathcal{K} && \text{(4.19b)}\\
\int_{\Theta} d_{\gamma,k}^2\; dP(d) \;&=\; \mu_{\gamma,k}^2 + \sigma_{\gamma,k}^2,\;\; \forall \gamma \in \Gamma, k \in \mathcal{K}\Big\}. && \text{(4.19c)}
\end{aligned}$$

$\Phi(\boldsymbol{\mu}, \boldsymbol{\sigma}, \Theta)$ is the set of all plausible surgery distributions that satisfy constraints (4.19a)-(4.19c). The constraint (4.19a) ensures that this ambiguity set contains only plausible probability distributions over the polyhedral support set Θ. The constraints (4.19b)-(4.19c) limit such probability distributions to have marginal first and second distributional moments being equal to those of the observed surgery durations. This moment-based ambiguity set satisfies all candidate distributions whose marginal means and standard deviations match $\mu_{\gamma,k}$ and $\sigma_{\gamma,k}$, respectively, for each pair of patient class $\gamma \in \Gamma$ and surgeon $k \in \mathcal{K}$.

We are now ready to develop the IMSDRO model for the CAS problem, which is derived based on both Assumptions 1 and 2. The integrated model combines the MS-MIP model (4.1)-(4.17) with a DRO approach such that we can handle different types of uncertainty in one optimization model. We formulate this integrated model as the following *min-max*

problem:

$$Z^{IMSDRO} = \min_{\boldsymbol{x},\boldsymbol{y},\widehat{\boldsymbol{y}},\boldsymbol{q}} \sum_{s\in\mathcal{S}} \pi_s \left\{ \sum_{k\in\mathcal{K}} \sum_{m\in\mathcal{L}} q^k_{m,s} + \max_{P\in\Phi(\boldsymbol{\mu},\boldsymbol{\sigma},\Theta)} \mathbb{E}_P\Big[f_s(\boldsymbol{y_s},\widehat{\boldsymbol{y}},\boldsymbol{d})\Big] \right\} \tag{4.20a}$$

$$\text{s.t. } (\boldsymbol{x_s},\ \boldsymbol{y_s},\widehat{\boldsymbol{y}},\ \boldsymbol{q_s}) \in \mathcal{R}_s,\ \ \forall s\in\mathcal{S} \tag{4.20b}$$

where $\mathbb{E}_P$ is the expectation taken over the probability distribution P, and the feasible region $\mathcal{R}_s$ is defined by constraints (4.4)-(4.10) and (4.12)-(4.17) for each individual scenario $s \in \mathcal{S}$. Given the surgery appointment decisions $\boldsymbol{y_s}$ and $\widehat{\boldsymbol{y}}$, and a realization of random variable $\boldsymbol{d}$, $f_s(\boldsymbol{y_s},\widehat{\boldsymbol{y}},\boldsymbol{d})$ is defined by $\sum_{n\in\mathcal{L}}\sum_{k\in\mathcal{K}} \max\Big\{0, \sum_{\gamma\in\Gamma} d_{\gamma,k}\Big(\sum_{t\in\mathcal{U}}\sum_{m\in\mathcal{T}\setminus\{t_0\}} \widetilde{y}^{k,n}_{\gamma,t,m} + \sum_{t\in\mathcal{T}}\sum_{m\in\mathcal{T}\setminus\{t_0\}} y^{k,n}_{\gamma,t,m,s} + \sum_{t\in\mathcal{T}\cup\{t_0\}} \widehat{y}^{k,n}_{\gamma,t,t_0}\Big) - V^k_n\Big\}$.

Intuitively, $f_s(\boldsymbol{y_s},\widehat{\boldsymbol{y}},\boldsymbol{d})$ is the cumulative surgical overtimes of all surgeons over the scheduling horizon. The objective function of the IMSDRO model (4.20a)-(4.20b) then implies that we are making the clinic and surgery appointment decisions, and clinical and surgical overtimes decisions so as to minimize the expected clinical overtimes plus the *worst-case* expected surgical overtimes of all surgeons over the set of plausible surgery duration distributions $P \in \Phi(\boldsymbol{\mu},\boldsymbol{\sigma},\Theta)$. The distributionally robust part seeks the worst-case distribution P of $\boldsymbol{d}$ for which $\mathbb{E}_P\big[f_s(\boldsymbol{y_s},\widehat{\boldsymbol{y}},\boldsymbol{d})\big]$ is maximized.

Our next step is to reformulate the min-max IMSDRO model (4.20a)-(4.20b) into a *tractable* reformulation using the moment-based ambiguity set $\Phi(\boldsymbol{\mu},\boldsymbol{\sigma},\Theta)$. To this aim, we first analyze the inner maximization problem in the IMSDRO model (4.20a)-(4.20b). For any fixed surgical decisions $\boldsymbol{y_s}$ and $\widehat{\boldsymbol{y}}$, and the uncertain realization vector $\boldsymbol{d}$, we consider the following *moment problem*:

$$\max_{P\in\Phi(\boldsymbol{\mu},\boldsymbol{\sigma},\Theta)} \mathbb{E}_P\Big[f_s(\boldsymbol{y_s},\widehat{\boldsymbol{y}},\boldsymbol{d})\Big]. \tag{4.21}$$

The decision variable in the moment problem (4.21) is the probability measure P in the ambiguity set $\Phi(\boldsymbol{\mu},\boldsymbol{\sigma},\Theta)$. To analyze this problem, we first expand this moment problem, which helps convert the min-max IMSDRO model (4.20a)-(4.20b) into an equivalent single-level minimization problem.

Remark. The min-max IMSDRO model (4.20a)-(4.20b) with the moment problem (4.21) has a special structure, which is useful for many operational problems in which $f_s(\boldsymbol{y_s},\widehat{\boldsymbol{y}},\boldsymbol{d})$ has the form of cumulative maximization values. So, our methodologies can be used for a broad range of settings.

Proposition 11 (Reformulation of the min-max IMSDRO Model). Under the moment-based ambiguity set $\Phi(\boldsymbol{\mu}, \boldsymbol{\sigma}, \Theta)$ for the probability distribution P of the surgery duration characterized by the constraints (4.19a)-(4.19c), the "min-max" IMSDRO model (4.20a)-(4.20b) can be reformulated as the following equivalent "minimization" problem,

$$Z^{IMSDRO} = \min_{\boldsymbol{x},\boldsymbol{y},\widehat{\boldsymbol{y}},\boldsymbol{q},\boldsymbol{\delta},\boldsymbol{\alpha},\boldsymbol{\beta}} \sum_{s\in\mathcal{S}} \pi_s \left\{ \sum_{k\in\mathcal{K}} \left(\sum_{m\in\mathcal{L}} q^k_{m,s} + \sum_{\gamma\in\Gamma} \left(\mu_{\gamma,k}\ \alpha^k_{\gamma,s} + (\mu^2_{\gamma,k} + \sigma^2_{\gamma,k})\ \beta^k_{\gamma,s} \right) \right) + \delta_s \right\} \tag{4.22a}$$

$$\text{s.t. } \delta_s \geq \max_{\boldsymbol{d}\in\Theta} \left\{ f_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{d}) - \sum_{\gamma\in\Gamma}\sum_{k\in\mathcal{K}} d_{\gamma,k}\ \alpha^k_{\gamma,s} - \sum_{\gamma\in\Gamma}\sum_{k\in\mathcal{K}} d^2_{\gamma,k}\ \beta^k_{\gamma,s} \right\}, \quad \forall s \in \mathcal{S} \tag{4.22b}$$

$$(\boldsymbol{x_s},\ \boldsymbol{y_s}, \widehat{\boldsymbol{y}},\ \boldsymbol{q_s}) \in \mathcal{R}_s, \quad \forall s \in \mathcal{S} \tag{4.22c}$$

$$\delta_s \in \mathbb{R},\ \boldsymbol{\alpha_s},\ \boldsymbol{\beta_s} \in \mathbb{R}^{|\Gamma|\times|\mathcal{K}|}, \quad \forall s \in \mathcal{S}, \tag{4.22d}$$

where $\delta_s \in \mathbb{R}$, and $\boldsymbol{\alpha_s}, \boldsymbol{\beta_s} \in \mathbb{R}^{|\Gamma|\times|\mathcal{K}|}$ are dual variables for constraints (4.19a)-(4.19c), respectively.

Structural properties. The proof of Proposition 11 is provided in Appendix A.1. The reformulation (4.22a)-(4.22d) of the IMSDRO model obtained in Proposition 11 is still nonlinear due to the maximization expression on the right-hand side of the constraint (4.22b). To obtain a tractable reformulation, we first attain a characterization of the overtime function $f_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{d})$ by converting it into an equivalent minimization linear program (LP) with the help of the surgical overtime definition for each surgeon (see the LP (4.34a)-(4.34c) in the proof of Proposition 12 in Appendix A.2). We formulate its dual problem in order to merge it with the maximization over $\boldsymbol{d} \in \Theta$ in the constraint (4.22b), and then reformulate the resulting problem based on the special structural properties, including (i) the surgery duration $\boldsymbol{d}$ has a *polyhedron-shaped support* Θ, and (ii) the dual variables for the $f_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{d})$ problem is *bounded* below and above by zero and one.

Proposition 12 (Reformulation of surgical overtime function). For any fixed and feasible value of $\boldsymbol{y_s}$, $\widehat{\boldsymbol{y}}$, $\boldsymbol{\alpha_s}$, $\boldsymbol{\beta_s}$, $\boldsymbol{x_s}$, and $\boldsymbol{q_s}$ vectors and δ_s under scenario $s \in \mathcal{S}$ in the minimization problem (4.22a)-(4.22d), the value of the maximization problem on the right hand side of constraint (4.22b), i.e., $\Psi_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s}) = \max_{\boldsymbol{d}\in\Theta} \{ f_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{d}) - \sum_{\gamma\in\Gamma}\sum_{k\in\mathcal{K}} d_{\gamma,k}\alpha^k_{\gamma,s} -$

$\sum_{\gamma\in\Gamma}\sum_{k\in\mathcal{K}} d_{\gamma,k}^2\beta_{\gamma,s}^k\}$, is equivalent to the following problem under each scenario $s\in\mathcal{S}$:

$$\Psi_s(\boldsymbol{y_s},\widehat{\boldsymbol{y}},\boldsymbol{\alpha_s},\boldsymbol{\beta_s}) = \max_{\lambda_s\in\Lambda_s}\sum_{\gamma\in\Gamma}\sum_{k\in\mathcal{K}}\Bigg\{\max_{d_{\gamma,k}^{LB}\le d_{\gamma,k}\le d_{\gamma,k}^{UB}}\Bigg(\sum_{n\in\mathcal{L}}\Big\{\sum_{t\in\mathcal{U}}\sum_{m\in\mathcal{T}\setminus\{t_0\}}\widetilde{y}_{\gamma,t,m}^{k,n}+ \tag{4.23}$$

$$\sum_{t\in\mathcal{T}}\sum_{m\in\mathcal{T}\setminus\{t_0\}} y_{\gamma,t,m,s}^{k,n}+\sum_{t\in\mathcal{T}\cup\{t_0\}}\widehat{y}_{\gamma,t,t_0}^{k,n}\Big\}\cdot\lambda_{n,s}^k\, d_{\gamma,k}-\alpha_{\gamma,s}^k\, d_{\gamma,k}-\beta_{\gamma,s}^k\, d_{\gamma,k}^2\Bigg)-\sum_{n\in\mathcal{L}}V_n^k\lambda_{n,s}^k\Bigg\},$$

where feasible region Λ_s is a polyhedron given by $\Lambda_s=\{\boldsymbol{\lambda_s}\in\mathbb{R}^{|\mathcal{K}|\times|\mathcal{L}|}: 0\le\lambda_{n,s}^k\le 1,\forall k\in\mathcal{K},n\in\mathcal{L}\}$ for each scenario $s\in\mathcal{S}$, and $\lambda_{n,s}^k$ is the dual variable associated with surgical overtime constraints.

The proof of Proposition 12 is provided in Appendix A.2.

Discrete approximations. We next analyze the inner-maximization problem (4.24) embedded in $\Psi_s(\boldsymbol{y_s},\widehat{\boldsymbol{y}},\boldsymbol{\alpha_s},\boldsymbol{\beta_s})$ for each pair of class $\gamma\in\Gamma$ and surgeon $k\in\mathcal{K}$ and compute its optimal solution based on the structure of the polyhedron-shaped support Θ defined by the set (4.18):

$$\max_{d_{\gamma,k}^{LB}\le d_{\gamma,k}\le d_{\gamma,k}^{UB}}\Bigg(\sum_{n\in\mathcal{L}}\Big\{\sum_{t\in\mathcal{U}}\sum_{m\in\mathcal{T}\setminus\{t_0\}}\widetilde{y}_{\gamma,t,m}^{k,n}+\sum_{t\in\mathcal{T}}\sum_{m\in\mathcal{T}\setminus\{t_0\}} y_{\gamma,t,m,s}^{k,n} \tag{4.24}$$

$$+\sum_{t\in\mathcal{T}\cup\{t_0\}}\widehat{y}_{\gamma,t,t_0}^{k,n}\Big\}\cdot\lambda_{n,s}^k\, d_{\gamma,k}-\alpha_{\gamma,s}^k\, d_{\gamma,k}-\beta_{\gamma,s}^k\, d_{\gamma,k}^2\Bigg).$$

The inner-maximization problem (4.24) is a concave quadratic program. However, finding a closed-form solution for this problem over $d_{\gamma,k}$ is not trivial because the optimal value of $d_{\gamma,k}$ depends on its coefficients in the inner-maximization problem (4.24) (which are themselves variables in the problem (4.23)). Even if the closed-form optimal solution for $d_{\gamma,k}$ is incorporated into the objective function (4.23), it becomes non-linear because we obtain a quadratic expression in $\lambda_{n,s}^k$. To overcome this issue, we approximate the inner-maximization problem (4.24) using a *piece-wise linear function* with equal length pieces. This is a common technique in optimization [163]. We define a set of $H+1$ segment points $\Upsilon_{\gamma,k}=\{\tilde{d}_{\gamma,k}(i)\}_{i=0}^H$ for the surgery duration of each patient class $\gamma\in\Gamma$ and surgeon $k\in\mathcal{K}$ pair, where $\tilde{d}_{\gamma,k}(i)=(1-\frac{i}{H})\, d_{\gamma,k}^{LB}+(\frac{i}{H})\, d_{\gamma,k}^{UB}$, $i\in\{0,\cdots,H\}$ is the i^{th} *segment point* in the set $\Upsilon_{\gamma,k}$ for each patient class $\gamma\in\Gamma$ and surgeon $k\in\mathcal{K}$ pair.

The inner-maximization problem (4.24) then reduces to the following approximation problem of finding the maximum over $H+1$ different quantities for each (γ,k) pair under

each scenario s:

$$\max_{i=0,\cdots,H} \Bigg\{ \Bigg(\sum_{n\in\mathcal{L}} \Big\{ \sum_{t\in\mathcal{U}} \sum_{m\in\mathcal{T}\setminus\{t_0\}} \widetilde{y}^{k,n}_{\gamma,t,m} \lambda^k_{n,s} + \sum_{t\in\mathcal{T}} \sum_{m\in\mathcal{T}\setminus\{t_0\}} y^{k,n}_{\gamma,t,m,s} \lambda^k_{n,s} + \sum_{t\in\mathcal{T}\cup\{t_0\}} \widehat{y}^{k,n}_{\gamma,t,t_0} \lambda^k_{n,s} \Big\} \Bigg) \tilde{d}_{\gamma,k}(i) - \alpha^k_{\gamma,s}\, \tilde{d}_{\gamma,k}(i) - \beta^k_{\gamma,s}\, \tilde{d}_{\gamma,k}(i)^2 \Bigg\}. \tag{4.25}$$

Note that choosing a large number of segment points for each (γ, k) pair models the support of the surgery duration distribution more precisely, thereby increasing the precision of the estimation made by the approximation problem (4.25) for the inner-maximization problem (4.24); however, this comes at the cost of more computational time. In §4.6.4, we investigate how different choices of segment points affect the solution quality and computational time. We demonstrate that our approach results in a more accurate objective function approximation as the number of segment points increases while computational time grows slower than linearly.

If we insert the approximation problem (4.25) into the optimization problem (4.23) derived in Proposition 12 under each scenario $s \in \mathcal{S}$, it yields an approximation called $\widetilde{\Psi}_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s})$ for the problem (4.23). In Theorem IV.1, we find an equivalent *mixed-integer linear program* for the approximation problem $\widetilde{\Psi}_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s})$ by leveraging McCormick-type constraints [125].

Theorem IV.1 (Scenario cut-generating problem)**.** *Under each scenario $s \in \mathcal{S}$, the optimization problem (4.23) is approximated by the following mixed-integer linear program (MILP):*

$$\widetilde{\Psi}_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s}) = \max_{\boldsymbol{\tau},\boldsymbol{\eta},\boldsymbol{\lambda}} \ \chi_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s}; \boldsymbol{\tau_s}, \boldsymbol{\eta_s}, \boldsymbol{\lambda_s}) \tag{4.26a}$$

$$s.t. \ \sum_{i=0}^{H} \eta^k_{\gamma,i} \ = \ 1, \ \ \forall \gamma \in \Gamma, k \in \mathcal{K} \tag{4.26b}$$

$$\tau^k_{n,s,\gamma,i} - \lambda^k_{n,s} - \eta^k_{\gamma,i} \ \geq \ -1, \ \forall \gamma \in \Gamma, k \in \mathcal{K}, n \in \mathcal{L}, i = 0, \cdots, H \tag{4.26c}$$

$$\tau^k_{n,s,\gamma,i} - \lambda^k_{n,s} \ \leq \ 0, \quad \forall \gamma \in \Gamma, k \in \mathcal{K}, n \in \mathcal{L}, i = 0, \cdots, H \tag{4.26d}$$

$$\tau^k_{n,s,\gamma,i} - \eta^k_{\gamma,i} \ \leq \ 0, \ \ \forall \gamma \in \Gamma, k \in \mathcal{K}, n \in \mathcal{L}, i = 0, \cdots, H \tag{4.26e}$$

$$\tau^k_{n,s,\gamma,i} \geq 0, \ \eta^k_{\gamma,i} \in \{0,1\}, \ 0 \leq \lambda^k_{n,s} \leq 1, \ \ \forall \gamma \in \Gamma, k \in \mathcal{K}, n \in \mathcal{L}, i = 0, \cdots, H \tag{4.26f}$$

where the objective function $\chi_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s}; \boldsymbol{\tau_s}, \boldsymbol{\eta_s}, \boldsymbol{\lambda_s})$ is defined for each scenario $s \in \mathcal{S}$ as

follows:

$$\sum_{\gamma\in\Gamma}\sum_{k\in\mathcal{K}}\sum_{i=0}^{H}\Bigg\{\sum_{n\in\mathcal{L}}\Big(\sum_{t\in\mathcal{U}}\sum_{m\in\mathcal{T}\setminus\{t_0\}}\widetilde{y}^{k,n}_{\gamma,t,m}+\sum_{t\in\mathcal{T}}\sum_{m\in\mathcal{T}\setminus\{t_0\}}y^{k,n}_{\gamma,t,m,s}+\sum_{t\in\mathcal{T}\cup\{t_0\}}\widehat{y}^{k,n}_{\gamma,t,t_0}\Big)\,\tilde{d}_{\gamma,k}(i)\,\tau^{k}_{n,s,\gamma,i} \tag{4.27}$$

$$-\,\alpha^{k}_{\gamma,s}\,\tilde{d}_{\gamma,k}(i)\,\eta^{k}_{\gamma,i}-\beta^{k}_{\gamma,s}\,\tilde{d}_{\gamma,k}(i)^2\,\eta^{k}_{\gamma,i}\Bigg\}-\sum_{k\in\mathcal{K}}\sum_{n\in\mathcal{L}}V^{k}_{n}\,\lambda^{k}_{n,s}.$$

The proof of Theorem IV.1 is provided in Appendix A.3. The important implication of Theorem IV.1 is that it prevents having an embedded MILP model on the right-hand side of the constraints (4.22b) by recognizing which scenario cuts must be added to replace the nonlinear constraints (4.22b). Using the results of Theorem IV.1, we can approximate the IMSDRO model (4.22a)-(4.22d) as follows:

$$\widetilde{Z}^{IMSDRO}=\min_{\boldsymbol{x},\boldsymbol{y},\widehat{\boldsymbol{y}},\boldsymbol{q},\boldsymbol{\delta},\boldsymbol{\alpha},\boldsymbol{\beta}}\sum_{s\in\mathcal{S}}\pi_s\Bigg\{\sum_{k\in\mathcal{K}}\Big(\sum_{m\in\mathcal{L}}q^{k}_{m,s}+\sum_{\gamma\in\Gamma}\Big(\mu_{\gamma,k}\,\alpha^{k}_{\gamma,s}+(\mu^2_{\gamma,k}+\sigma^2_{\gamma,k})\,\beta^{k}_{\gamma,s}\Big)\Big)+\delta_s\Bigg\} \tag{4.28a}$$

$$\text{s.t. }\ \delta_s\ \geq\ \widetilde{\Psi}_s(\boldsymbol{y_s},\widehat{\boldsymbol{y}},\boldsymbol{\alpha_s},\boldsymbol{\beta_s}),\ \ \forall s\in\mathcal{S} \tag{4.28b}$$

$$(\boldsymbol{x_s},\ \boldsymbol{y_s},\widehat{\boldsymbol{y}},\ \boldsymbol{q_s})\ \in\ \mathcal{R}_s,\ \ \forall s\in\mathcal{S} \tag{4.28c}$$

$$\delta_s\ \in\ \mathbb{R},\ \boldsymbol{\alpha_s},\ \boldsymbol{\beta_s}\ \in\ \mathbb{R}^{|\Gamma|\times|\mathcal{K}|},\ \ \forall s\in\mathcal{S}. \tag{4.28d}$$

Remark. The minimization problem (4.28a)-(4.28d) is an approximation of the IMSDRO model (4.22a)-(4.22d). We shall call it the IMSDRO-APRX model. Although IMSDRO-APRX model (4.28a)-(4.28d) has a linear objective function with continuous and binary variables, due to the right-hand side of constraints (4.28b), which includes an embedded optimization problem (4.26a)-(4.26f), this model is not an MILP that is solvable by off-the-shelf MILP solvers (such as Gurobi and Cplex). In §4.4, we develop a constraint generation algorithm, which is based on iteratively generating constraints (4.28b) for each individual scenario $s\in\mathcal{S}$, as needed, to efficiently solve the IMSDRO-APRX model.

4.4 Constraint Generation Algorithm

In this section, we describe a new constraint generation algorithm for solving the IMSDRO-APRX model, which exploits the structure of the embedded MILP $\widetilde{\Psi}_s(\boldsymbol{y_s},\widehat{\boldsymbol{y}},\boldsymbol{\alpha_s},\boldsymbol{\beta_s})$ to generate effective scenario cuts. The main idea of this algorithm is explained as follows. The

algorithm starts by solving the IMSDRO-APRX model without having any of the constraints (4.28b). At each iteration, the algorithm solves a *relaxed master problem* (RMP) to obtain a solution $(\boldsymbol{x_s}, \boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{q_s}, \delta_s, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s})$. Given this solution, it then solves what we call the *scenario cut-generating problem* $\widetilde{\Psi}_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s})$ or (4.26a)-(4.26f). If $\widehat{\boldsymbol{y}}, \boldsymbol{y_s}, \boldsymbol{\alpha_s}$, and $\boldsymbol{\beta_s}$ do not satisfy $\delta_s \geq \widetilde{\Psi}_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s})$, the scenario cut-generating problem returns scenario cuts in the form of (4.29b) back to RMP and the algorithm proceeds to next iteration. If $(\boldsymbol{x_s}, \boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{q_s}, \delta_s, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s})$ is optimal, the algorithm then terminates. The RMP at the R^{th} iteration is formulated as follows,

$$\widetilde{Z}^{RMP} = \min_{\boldsymbol{x},\boldsymbol{y},\widehat{\boldsymbol{y}},\boldsymbol{q},\boldsymbol{\delta},\boldsymbol{\alpha},\boldsymbol{\beta}} \sum_{s\in\mathcal{S}} \pi_s \left\{ \sum_{k\in\mathcal{K}} \left(\sum_{m\in\mathcal{L}} q^k_{m,s} + \sum_{\gamma\in\Gamma} \left(\mu_{\gamma,k}\ \alpha^k_{\gamma,s} + (\mu^2_{\gamma,k} + \sigma^2_{\gamma,k})\ \beta^k_{\gamma,s} \right) \right) + \delta_s \right\} \tag{4.29a}$$

$$\text{s.t. } \delta_s \geq \chi_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s}; \boldsymbol{\tau_s^{(r)}}, \boldsymbol{\eta_s^{(r)}}, \boldsymbol{\lambda_s^{(r)}}), \quad \forall s \in \mathcal{S}, r = 1, \cdots, R-1 \tag{4.29b}$$

$$(\boldsymbol{x_s}, \boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{q_s}) \in \mathcal{R}_s, \quad \forall s \in \mathcal{S} \tag{4.29c}$$

$$\delta_s \in \mathbb{R}, \ \boldsymbol{\alpha_s}, \boldsymbol{\beta_s} \in \mathbb{R}^{|\Gamma|\times|\mathcal{K}|}, \quad \forall s \in \mathcal{S}, \tag{4.29d}$$

where the superscript r is used to denote all the iterations up to the current iteration R, and the solution of the scenario cut-generating problem (4.26a)-(4.26f) at the r^{th} iteration is denoted by $(\boldsymbol{\tau_s^{(r)}}, \boldsymbol{\eta_s^{(r)}}, \boldsymbol{\lambda_s^{(r)}})$ for each scenario $s \in \mathcal{S}$. The linear function $\chi_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s}; \boldsymbol{\tau_s^{(r)}}, \boldsymbol{\eta_s^{(r)}}, \boldsymbol{\lambda_s^{(r)}})$ in the constraints (4.29b) is represented by the expression (4.27) with $(\boldsymbol{\tau_s}, \boldsymbol{\eta_s}, \boldsymbol{\lambda_s}) = (\boldsymbol{\tau_s^{(r)}}, \boldsymbol{\eta_s^{(r)}}, \boldsymbol{\lambda_s^{(r)}})$ for each scenario $s \in \mathcal{S}$. These scenario cuts are iteratively derived by passing the current solution $(\boldsymbol{x_s^{(R)}}, \boldsymbol{y_s^{(R)}}, \widehat{\boldsymbol{y}}^{(R)}, \boldsymbol{q_s^{(R)}}, \delta_s^{(R)}, \boldsymbol{\alpha_s^{(R)}}, \boldsymbol{\beta_s^{(R)}})$ of the RMP to the scenario cut-generating problem, and checking whether it satisfies (4.28b). If not, we add the corresponding scenario cut to the RMP for each scenario s. The details of the constraint generation algorithm are presented in Algorithm 4.

Our algorithm is similar to the L-shaped decomposition methods [32], where a large-scale stochastic model is solved by decomposing it into a master problem and sub-problems, and feasibility and optimality cuts are added once needed. The difference is our algorithm can be viewed as a *row generation* algorithm, because new constraints are added throughout a scenario cut-generating MILP. Note that the total number of scenario cuts that are being passed back to the RMP at each iteration is not necessarily equal to the total number of scenarios. Indeed, for only violated scenarios, Algorithm 4 passes back their corresponding scenario cuts to the RMP. In §4.6.4, we compare multi-cut and single-cut versions of Algorithm 4 for the CAS problem.

Theorem IV.2. *The constraint generation Algorithm 4 converges to an optimal scheduling*

Algorithm 4 Constraint Generation Algorithm for Solving the IMSDRO-APRX Model

Initialize iteration number $R = 0$, a positive tolerance ϵ, and also set the scenario parameters $Terminate(s) \leftarrow true$ for each scenario $s \in \mathcal{S}$.
Step I: **Solve the Relaxed Master Problem (RMP)**.
while ($\exists$ at least one $Terminate(s) \leftarrow true$ for a scenario $s \in \mathcal{S}$) **do**
    Set $R \leftarrow R + 1$, and $Terminate(s) \leftarrow false$ for each scenario $s \in \mathcal{S}$.
    Solve the RMP to get optimal solution $(\boldsymbol{x_s^{(R)}}, \boldsymbol{y_s^{(R)}}, \widehat{\boldsymbol{y}}^{(R)}, \boldsymbol{q_s^{(R)}}, \delta_s^{(R)}, \boldsymbol{\alpha_s^{(R)}}, \boldsymbol{\beta_s^{(R)}})$ for all $s \in \mathcal{S}$.
    Step II: **Cut-Generating Subroutine**.
    for each scenario $s \in \mathcal{S}$ **do**
        Solve the scenario cut-generating problem $\widetilde{\Psi}_s(\boldsymbol{y_s^{(R)}}, \widehat{\boldsymbol{y}}^{(R)}, \boldsymbol{\alpha_s^{(R)}}, \boldsymbol{\beta_s^{(R)}})$.
        Obtain the optimal solution $(\boldsymbol{\tau_s^{(R)}}, \boldsymbol{\eta_s^{(R)}}, \boldsymbol{\lambda_s^{(R)}})$.
        Step III: **Add Scenario Cuts to the RMP**.
        if $\delta_s^{(R)} < (1-\epsilon)\, \chi_s(\boldsymbol{y_s^{(R)}}, \widehat{\boldsymbol{y}}^{(R)}, \boldsymbol{\alpha_s^{(R)}}, \boldsymbol{\beta_s^{(R)}}; \boldsymbol{\tau_s^{(R)}}, \boldsymbol{\eta_s^{(R)}}, \boldsymbol{\lambda_s^{(R)}})$ **then**
            Add a scenario cut $\delta_s \geq \chi_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s}; \boldsymbol{\tau_s}, \boldsymbol{\eta_s}, \boldsymbol{\lambda_s})$ to the RMP for scenario $s \in \mathcal{S}$.
            Set $Terminate(s) \leftarrow true$ for scenario $s \in \mathcal{S}$.
Step IV: **Return the optimal policy**.
Return $(\boldsymbol{x_s^{(R)}}, \boldsymbol{y_s^{(R)}}, \widehat{\boldsymbol{y}}^{(R)}, \boldsymbol{q_s^{(R)}}, \delta_s^{(R)}, \boldsymbol{\alpha_s^{(R)}}, \boldsymbol{\beta_s^{(R)}})$ for each $s \in \mathcal{S}$ as the optimal policy.

policy for the IMSDRO-APRX model in a finite number of iterations.

The proof of Theorem IV.2 is provided in Appendix A.4.

4.5 Data-Driven Rolling Horizon Procedure

In general, the scheduling policy obtained from solving the IMSDRO-APRX model is not readily implementable for the real-world CAS problems because it is scenario dependent and does not allow for information gained over time to be used. Indeed, the critical limitation of scenario-based stochastic programs is that their optimal policy is only valid for a limited set of scenarios. To resolve this issue, we develop a new data-driven Rolling Horizon Procedure (RHP) to (i) make the scheduling policy *implementable* in practice, and (ii) evaluate the optimal scheduling policy empirically. In other words, it allows practitioners to make use of the latest data that is revealed as time progresses, and adjust their decisions in a rolling horizon framework. By using this data-driven RHP, we dynamically observe the realization of the uncertain parameters in one period and update the scenario tree for the following periods. The main goal is to evaluate the outcomes of implementing the optimal clinical/surgical decisions over a scheduling horizon on a rolling basis.

In our proposed data-driven RHP, one *sample path*, denoted by $\boldsymbol{\omega}$, is drawn from the historical data (see §4.6). This sample path includes the realized number of patient appoint-

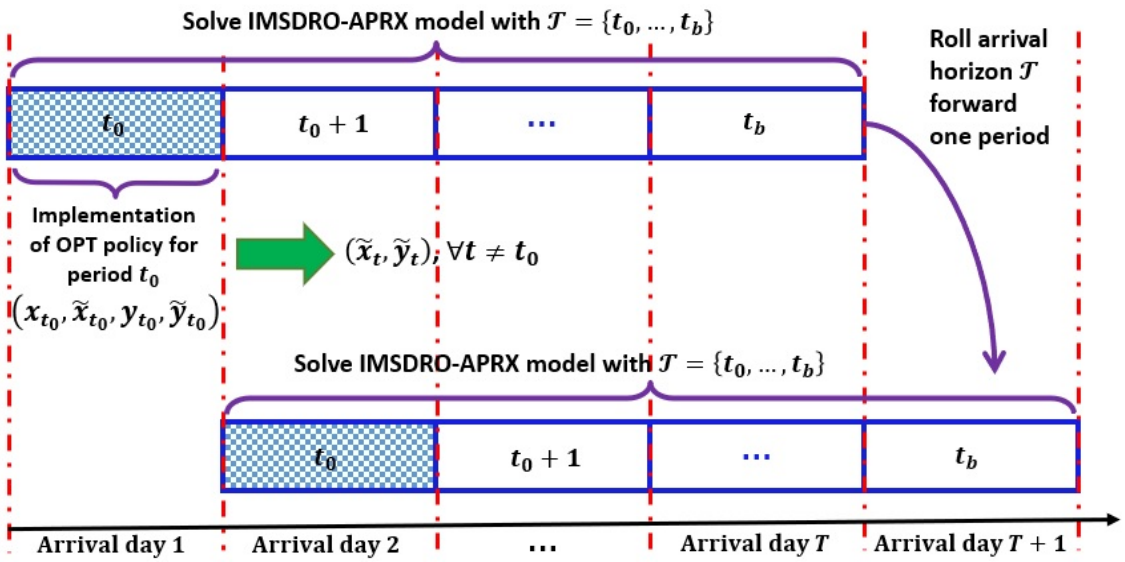


Figure 4.5: The illustration of the data-driven rolling horizon procedure for solving the IMSDR-APRX model with an arrival horizon of $\mathcal{T} = \{t_0, \cdots, t_b\}$ on every stage (day) for the CRS problem.

ment requests, the realized patients' surgery need, and the realized surgery durations over an arrival horizon of $T = |\mathcal{T}|$ periods (days). For implementation of a scheduling policy in the current period $t_0 \in \mathcal{T}$, given a scenario tree for the number of appointment requests over periods $t_0 + 1, \cdots, t_0 + T - 1$ and an ambiguity set for the surgery duration, we solve the T-stage IMSDRO-APRX model with an arrival horizon of T periods in which the uncertain parameters for period t_0 are known based on the realized path $\boldsymbol{\omega}$. We then implement the obtained optimal policy *only* for the current period t_0 and update the number of patients who need clinic and surgery appointments over $t_0 + 1, \cdots, t_0 + T - 1$ periods, as well as the remaining clinical and surgical capacities. We repeat this procedure, and "roll the patient arrival horizon forward one day" by adding a new period to the calendar at every step, so that at the following period $t_0 + 1$, the arrival horizon includes period $t_0 + 2$ to period $t_0 + T$. Note that the length of the arrival horizon is always T periods (see Figure 4.5). By drawing enough realized sample paths, we can estimate the average clinical and surgical overtimes of surgeons over all sample paths, and compare this objective function obtained by the data-driven RHP with the IMSDRO-APRX model's objective function value (with no rolling) (see §4.6.2). The data-driven RHP provides a framework that makes the decisions made by the IMSDRO approach implementable in practice.

The details of the data-driven RHP are presented in Algorithm 5. Here, IMSDRO-APRX$(i, \boldsymbol{\omega})$ represents the problem in which the first period of the arrival horizon is day i, and its data is based on the sample path $\boldsymbol{\omega}$ on the realization of arrival number $(\mathcal{D}_{\gamma,i}(\boldsymbol{\omega}))$,

Algorithm 5 Data-driven Rolling Horizon Procedure for the IMSDRO-APRX Model

1: **Step I: Initialization**. Consider a sample path $\boldsymbol{\omega}$ for realized number of appointment requests ($\mathcal{D}_{\gamma,i}(\boldsymbol{\omega})$), realizations of surgery duration ($d_{\gamma,k}(\boldsymbol{\omega})$), and the realized surgery requests ($\widetilde{z}^k_{\gamma,i}(\boldsymbol{\omega})$) for periods $i \in \mathcal{T} = \{t_0, \cdots, t_b\}$, and start at the beginning of arrival horizon (i.e., period $i = t_0$).

2: **Step II: Solve the IMSDRO-APRX model for each period** i.

3: **for** each arrival period $i \in \mathcal{T} = \{t_0, \cdots, t_b\}$ **do**

4: Consider a scenario tree with $|\mathcal{T}| - 1$ time periods.

5: Solve IMSDRO-APRX$(i, \boldsymbol{\omega})$ beginning with period i and parameters $\mathcal{D}_{\gamma,i}(\boldsymbol{\omega})$ and $\widetilde{z}^k_{\gamma,i}(\boldsymbol{\omega})$.

6: Given the following *implementable* decisions in period i for sample path $\boldsymbol{\omega}$,

$x^{k,m}_{p,\gamma,i}(\boldsymbol{\omega})$, where $k \in \mathcal{K}, \gamma \in \Gamma, m \in \mathcal{L}, p \in \mathcal{D}_{\gamma,i}(\boldsymbol{\omega})$, and i is an arrival day,

$\widetilde{x}^{k,i}_{p,\gamma,t}$, where $k \in \mathcal{K}, \gamma \in \Gamma, t \in \mathcal{U}, p \in \widetilde{\mathcal{D}}_{\gamma,t}$, and i is a clinic day,

$\widehat{y}^{k,n}_{\gamma,t,i}(\boldsymbol{\omega})$, where $k \in \mathcal{K}, \gamma \in \Gamma, t \in \mathcal{U} \cup \{i\}$, and $n \in \mathcal{L}$, and i is a clinic day

$\widetilde{y}^{k,i}_{\gamma,t,m}$, where $k \in \mathcal{K}, \gamma \in \Gamma, t \in \mathcal{U}, m \in \mathcal{U}$, and i is a surgery day,

calculate clinic and surgery overtimes ($q^k_i(\boldsymbol{\omega})$, $o^k_i(\boldsymbol{\omega})$) in period i for each $k \in \mathcal{K}$ as follows:

$$q^k_i(\boldsymbol{\omega}) = \left(\sum_{\gamma \in \Gamma} \Big(\sum_{t \in \mathcal{U}} \sum_{p \in \widetilde{\mathcal{D}}_{\gamma,t}} c_\gamma \cdot \widetilde{x}^{k,i}_{p,\gamma,t} + \sum_{t \in \mathcal{T}} \sum_{p \in \mathcal{D}^s_{\gamma,t}} c_\gamma \cdot x^{k,i}_{p,\gamma,i}(\boldsymbol{\omega}) \Big) \; - \; U^k_i \right)^+$$

$$o^k_i(\boldsymbol{\omega}) = \left(\sum_{\gamma \in \Gamma} \Big(\sum_{t \in \mathcal{U}} \sum_{m \in \mathcal{T} \setminus \{i\}} d_{\gamma,k}(\boldsymbol{\omega}) \cdot \widetilde{y}^{k,i}_{\gamma,t,m} + \sum_{t \in \mathcal{U} \cup \{i\}} d_{\gamma,k}(\boldsymbol{\omega}) \cdot \widehat{y}^{k,i}_{\gamma,t,i}(\boldsymbol{\omega}) \Big) - V^k_i \right)^+ .$$

7: Update the horizons $\mathcal{U}$, $\mathcal{T}$, and $\mathcal{L}$, and the following parameters for the next period $i + 1$:

$$\begin{aligned}
\widetilde{x}^{k,m-1}_{p,\gamma,t-1} &\leftarrow \widetilde{x}^{k,m}_{p,\gamma,t}, \text{ for } t < i,\ m > i. \\
\widetilde{x}^{k,m-1}_{p,\gamma,t-1} &\leftarrow x^{k,m}_{p,\gamma,t}(\boldsymbol{\omega}), \text{ for } t = i,\ m > i. \\
\widetilde{y}^{k,n-1}_{\gamma,t-1,m-1} &\leftarrow \widehat{y}^{k,n}_{\gamma,t,i}(\boldsymbol{\omega}), \text{ for } t < i,\ m = i,\ n > i. \\
\widetilde{y}^{k,n-1}_{\gamma,t-1,m-1} &\leftarrow \widetilde{y}^{k,n}_{\gamma,t,m}, \text{ for } t < i,\ m \leq i,\ n > i.
\end{aligned}$$

8: **Step III: Calculate the objective function for the sample path** $\boldsymbol{\omega}$.

$$Q(\boldsymbol{\omega}) \; = \; \sum_{i=t_0}^{t_b} Q_i(\boldsymbol{\omega}), \text{ where } Q_i(\boldsymbol{\omega}) \; = \; \sum_{k \in \mathcal{K}} \big(q^k_i(\boldsymbol{\omega}) + o^k_i(\boldsymbol{\omega}) \big).$$

surgery needs ($\widetilde{z}^k_{\gamma,i}(\boldsymbol{\omega})$) for $i \in \mathcal{T} = \{t_0, \cdots, t_b\}$, and surgery durations ($d_{\gamma,k}(\boldsymbol{\omega})$) for each patient class and surgeon. This sample path provides the related data for the new arrival horizon $\mathcal{T}$. Note that the next period, $i+1$, becomes the first period after rolling forward one period, and the constraint generation Algorithm 4 is used to solve IMSDRO-APRX$(i, \boldsymbol{\omega})$ again. After the T-stage IMSDRO-APRX$(i, \boldsymbol{\omega})$ model is solved, an i-th stage decision is implemented and new information is obtained, we roll forward (i.e., shift the time window) to solve another T-stage IMSDRO-APRX model with the uncertainty determined by the implemented i-th stage decision and by an observation of sample path $\boldsymbol{\omega}$ only for i-th stage. It is worth noting that the scenario tree that is considered in step 4 can be updated at each iteration i using the scenario generation and reduction algorithm illustrated in Appendix B, to capture any possible seasonality or trend in the data.

4.6 Case Study: Empirical Results and Managerial Insights

We populate our models and algorithms with appointment scheduling data from a highly specialized surgical clinic of a partner hospital. We provide numerical results to evaluate the performance of our models and algorithms compared to current practice of the surgical clinic. Our models and algorithms can be applied to similar settings in destination medical centers or typical hospitals to coordinate clinic and surgery visits given predefined priority-based access delay targets.

4.6.1 Experiment Setup

Appointment requests are received throughout the day either from other units within the same or nearby hospitals or from remote healthcare facilities. Each patient request asks for a clinic consultation appointment with one of the surgeons. Some patients may require a surgical procedure; this is determined during the patient's clinic consultation appointment. Appointment request forms include information on (i) the referral type, i.e., either local or remote referral, and (ii) the indication of disease, which can be one of the 12 possible medical conditions.

The surgical clinic is currently scheduling appointment requests with the surgeon who has the earliest clinic consultation availability. However, this policy has resulted in long wait time to surgery, which is particularly troubling for patients with acute conditions. Furthermore, it has been observed that some surgeons end their surgical day early on some days and very late on other days. Our models are designed to guarantee that all patients will be offered a clinic consultation and a surgical appointment (if needed) within a time window that is safe for them to wait. Our models minimize the clinical and surgical overtime incurred to provide

such access service level considering the uncertainty in request arrival and surgery duration and reduce the surgeon's end of day variability by leveraging our IMSDRO approach.

We consider five patient priority classes/types; class 1 includes the most acute and urgent conditions, whereas class 5 is assigned to patients who only need a clinic appointment for follow-up/consult or those who do not need a surgery in the near future. Each priority class corresponds to a maximum wait time to surgery WTS_γ, except class 5 that does not require a surgery. Clinic to surgery gap CSG_γ is another parameter that depends on patient priority class. The minimum wait time for clinic visits (WTC_γ), however, depends only on the referral type. For local referrals (i.e., the patient is physically at the hospital or in the same region), WTC_γ is zero, whereas for remote referrals, we assume WTC_γ is five business days from the day the request is received to give the patients at least one week to make travel arrangements. Table 4.2 shows WTC_γ, CSG_γ and WTS_γ values in days for different patient priority classes/types.

Class	**WTC**	**CSG**	**WTS**
1	0 or 5	0	5
2	0 or 5	1	7
3	0 or 5	2	10
4	0 or 5	3	18
5	0 or 5	NA	NA

Table 4.2: The values of Wait Time to Clinic (WTC), Clinic to Surgery Gap (CSG), and Wait Time to Surgery (WTS) in terms of number of days from our partner surgical hospital.

The probability of requiring a surgery procedure depends on the patient class. The surgical clinic under study performs about 400 surgeries per month. This corresponds to a rate of about 60 appointment requests per business day. Each clinic appointment takes 15 minutes in length; in other words, clinic days are divided into 15-minute time slots and each consultation appointment takes one slot. The surgical clinic has eight surgeons (i.e., $|\mathcal{K}| = 8$). These 8 surgeons are divided into two teams taking alternating turns between the clinic and the operating room (OR) from one day to another (i.e., on a given day, four surgeons are seeing patients in the clinic and four surgeons are performing surgeries in the OR). Therefore, each surgeon separately maintains both a clinical and a surgical calendar. The specialized surgical unit we model in our case study does have access to dedicated ORs as well as to a number of swing ORs that they can use, if needed. Thus, operating room capacity can be flexible, if needed.

Surgery duration $d_{\gamma,k}$ depends on both patient class and the specific surgeon who performs the operation. Recall that patient class γ is a tuple of two elements: referral type and indication of disease. We assume surgery duration is independent of referral type but depends

on the indication of disease. Therefore, given 8 surgeons and 12 disease indications, we consider 96 indication-surgeon pairs to define surgery duration. For each indication-surgeon combination, we employ the empirical surgery mean, standard deviation and support of past surgeries to construct the ambiguity sets for surgery duration. Appendix B elaborates on our approach for both generating the ambiguity set for the surgery duration and the scenario tree for the number of patient referrals. All algorithms are coded in Python 2.7.10 and the models are solved using Gurobi 5.6.3. The computations are performed on a Windows 7 machine with Core i7-2600M CPU at 3.40 GHz with 16 GB of memory.

4.6.2 Assessing the Performance of Different Scheduling Policies

We evaluate our stochastic-robust policy against four benchmark policies in terms of clinical and surgical overtimes (i.e., objective function) as well as clinical and surgical access times using the RHP proposed in §4.5. These four policies are summarized below.

- **Stochastic-robust policy**. This policy is obtained by solving the IMSDRO-APRX model (4.28a)-(4.28d), in which the uncertainty in the surgery duration is modeled by an ambiguity set, and the uncertainty in the number of appointment requests is modeled by a scenario tree.

- **Stochastic policy**. This policy is obtained by solving the MS-MIP model (4.1)-(4.17), in which the uncertainty in the number of appointment requests is modeled by a scenario tree, and the surgery durations are set to their empirical mean values.

- **Deterministic policy**. This scheduling policy is obtained by solving the deterministic version of the MS-MIP model (4.1)-(4.17), in which both surgery durations and the number of appointment requests are set to their empirical mean values.

- **Current policy**. This heuristic policy mimics the current/existing policy used by the surgical clinic. As discussed above, for each appointment request, this policy suggests the surgeon with the earliest clinic appointment availability. On the clinic appointment date, if the patient requires a surgery, it offers the earliest surgical appointment with the same surgeon.

The only stochastic element of the deterministic policy is the Bernoulli random variable governing whether the clinic visit reveals that a surgery is needed. Compared to this policy, the Poisson arrival process is added in the stochastic policy using a scenario tree. The stochastic-robust policy is the most realistic policy, building on the stochastic policy to incorporate the ambiguity set of surgery duration. To further evaluate our results, we deploy

two other instances A and B of the CAS problem in addition to our case study (base case). The differences between the case study and these instances are (i) the number of surgeons is 4 (instance A) and 10 (instance B) as opposed to 8 (case study), and (ii) the number of scenarios for arrival is 20 (instance A) and 10 (instance B) as opposed to 14 (case study). Other parameters of test instances A and B are similar to the case study.

Evaluation of Objective Function Values (Overtime). For this analysis, 60 sample paths of a length 5 working weekdays for the arrival horizon are randomly drawn that include the realized (i) number of patient appointment referrals, (ii) surgery needs, and (iii) surgery durations. For each sample path, the RHP (Algorithm 5) is deployed to implement the above policies that are obtained by solving the IMSDRO-APRX, MS-MIP and deterministic models. We consider a 5-day arrival horizon and roll the horizon forward to cover 5 days. Finally, we calculate the mean ($\tilde{Z}$) and standard deviation ($\tilde{\sigma}$) of the objective functions (clinical and surgical overtimes) over all sample paths as the output of the data-driven RHP to assess the policies. Note that we start our analyses with long-run average system state and further use a 10-day burn-in period so that our results and findings are not affected by the initial system status.

The empirical results for our case study (base case) as well as test instances A and B are reported in Table 4.3. The optimal objective values (Z^*) are calculated by solving the IMSDRO-APRX, MS-MIP and deterministic models (without using the RHP). The maximum ($\tilde{Z}_{\max}$) and the mean ($\tilde{Z}$) objective value and the standard deviations ($\tilde{\sigma}$) are the maximum, the mean and the standard deviation, respectively, of the true objective function values obtained by using the data-driven RHP on the 60 sample paths described above. The *out-of-sample stability error* is calculated as follows:

$$\text{Out-of-sample Stability Error} = \frac{|\text{Mean objective value } (\tilde{Z}) - \text{Optimal objective value } (Z^*)|}{\text{Optimal objective value } (Z^*)}.$$

We do not include the current policy in this analysis. This is because the current policy does not incorporate the access target constraints in its decision, and there is no optimal objective value for this policy. We do, however, incorporate it in the clinical and surgical access times analyses below.

As reported in Table 4.3, the *optimal objective function value* of the stochastic-robust policy is slightly larger (2.9% more overtime) than the stochastic policy. This is because the stochastic-robust policy adds significant uncertainty to the surgery duration by considering a whole range of possibilities for the probability distribution of the surgery duration and optimizes the *worst-case performance* as opposed to the stochastic policy that only considers the surgery duration mean and optimizes the *mean performance*. Also, the deterministic

Statistics	Stochastic-Robust Policy	Stochastic Policy	Deterministic Policy
The case study:			
Optimal objective value (Z^*)	4,451	4,325	4,311
Mean objective value ($\tilde{Z}$)	4,317	4,525	4,788
Max objective value ($\tilde{Z}_{\max}$)	5,285	5,389	5,528
Standard deviation ($\tilde{\sigma}$)	378	416	445
Out-of-sample error	3.10%	4.42%	9.96%
Test instance A:			
Optimal objective value (Z^*)	6,375	6,211	6,125
Mean objective value ($\tilde{Z}$)	6,254	6,348	6,633
Max objective value ($\tilde{Z}_{\max}$)	6,835	6,992	7,025
Standard deviation ($\tilde{\sigma}$)	656	712	695
Out-of-sample error	1.93%	2.16%	7.66%
Test instance B:			
Optimal objective value (Z^*)	2,274	2,175	2,165
Mean objective value ($\tilde{Z}$)	2,199	2,257	2,379
Max objective value ($\tilde{Z}_{\max}$)	2,868	2,912	3,012
Standard deviation ($\tilde{\sigma}$)	295	318	342
Out-of-sample error	2.99%	3.63%	9.00%

Table 4.3: The out-of-sample stability analysis of the stochastic-robust, stochastic, and deterministic policies in terms of objective function in the case study, and test instances A and B without and with the RHP Algorithm 5. Numbers are the total clinical and surgical overtime aggregated over all 8 surgeons and the 5-day horizon.

policy has the smallest optimal objective function value, because it optimizes for a single scenario in which the number of patient referrals and surgery durations are both set to their empirical means.

However, when we simulate and exploit these scheduling polices for the 60 sample paths, we observe from Table 4.3 that the stochastic-robust policy yields both the smallest *mean objective function value* (i.e. the lowest surgeon mean overtime) and the smallest variability around the overtime (i.e. the lowest standard deviation) relative to the stochastic and deterministic policies. In particular, the mean objective function value (i.e., overtime) of the stochastic-robust policy is 4.6% and 9.8% less than the stochastic and deterministic policies, respectively. This observation illustrates that even though the stochastic-robust policy hedges against the worst-case and makes *more conservative* decision strategies, the worst-case situation may not necessarily occur for all possible surgeries in practice. This can subsequently leads to a smaller overtime mean and variability for the stochastic-robust policy relative to the other policies. Limiting the variability is of paramount importance in healthcare appointment scheduling as having consistent performance allows the surgical clinic to better plan for and manage their resources. On the other hand, the stochastic policy makes scheduling decisions based on the surgery duration mean scenario; therefore, it results in a *more compact* schedule compared to the stochastic-robust policy. However, the surgery duration mean scenario does not necessarily happen for all possible surgeries in practice, which subsequently leads to more overtime relative to the stochastic-robust policy. The deterministic policy has the poorest performance with respect to both mean objective value and variability. This is because it deploys a policy that was optimized for only one single scenario of patient arrival mean and surgery duration mean, for many sample paths. Results are similar for the test instances A and B.

Moreover, the *out-of-sample stability* [96] guarantees that the mean objective function value $\tilde{Z}$ obtained from implementing the optimal scheduling policy by using the data-driven RHP is approximately the same as the optimal objective value Z^* of the stochastic models. As reported in Table 4.3, the stochastic-robust policy has the *smallest* out-of-sample stability error (3.10%) compared to the stochastic policy (4.42%) and the deterministic policy (9.96%) in the case study and similarly in the two test instances A and B. The small difference between the mean objective function value and the optimal objective value further confirms the validity and reliability of the IMSDRO-APRX model as a reasonable approximation method.

To further assess the stochastic-robust, stochastic, and deterministic policies, we implement the RHP for the described policies by solving the models with a 5-day arrival horizon and roll the horizon forward to cover 10 days. Note that our approach is not limited to

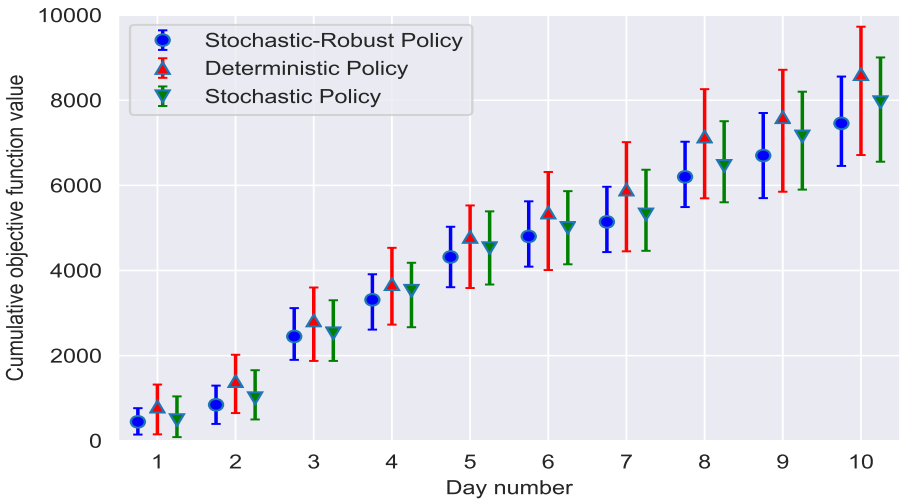


Figure 4.6: The comparison of the stochastic-robust, stochastic, and deterministic policies over 10 business days implemented by the RHP in terms of mean, 25%-QT and 75%-QT cumulative overtimes for the case study.

a 10-day roll-out window and one may continue rolling the horizon forward for as long as needed. We use this roll-out window for demonstration only. Figure 4.6 graphs the cumulative clinic and surgery overtime of these policies (aggregated over all 8 surgeons) over 10 business days for the case study. From Figure 4.6, we observe that the stochastic-robust policy is meaningfully better than both the stochastic and deterministic policies in terms of both cumulative overtime mean and variability, and this is consistent through the end of the horizon. Therefore, the stochastic-robust policy offers an excellent performance with a much lower variability, which is critical for implementation in everyday practice. Results were similar for test instances A and B and are not shown here.

Evaluation of Clinical and Surgical Access Times. In the next analysis, we compare the scheduling policies in terms of the clinical and surgical access times. To this aim, we define a new performance metric for earliness or tardiness that is "the number of days between the scheduled clinic and surgery appointment date and the maximum wait time target date." We call it the *access measure with respect to the target*. Since there are different patient classes with various access delay target windows, this metric helps us better understand how much earlier or later, with respect to the maximum wait target, the policies schedule the appointments. The negative (positive) value implies how much earlier (later) a policy schedule the clinic and surgery appointment with regard to the maximum wait target. In this analysis, we use the RHP for implementing the stochastic-robust, stochastic, and deterministic policies obtained by solving the IMSDRO-APRX, MS-MIP and deterministic models, respectively. Note that the current policy is also included in this analysis; it assigns

new appointment requests to the surgeon with the earliest availability. We calculate this access earliness or tardiness measure with respect to the target for each patient whose referral is received within the roll-out window for the case study as well as the test instances A and B. We compute the mean, worst-case, and standard deviation (SD) of these access measures.

Statistics	Clinical Access Measure with respect to the Target				Surgical Access Measure with respect to the Target			
	Stochastic-Robust Policy	Stochastic Policy	Deterministic Policy	Current Policy	Stochastic-Robust Policy	Stochastic Policy	Deterministic Policy	Current Policy
The case study:								
Mean	−2.42	−2.67	−2.97	−4.55	−2.78	−2.83	−2.97	4.65
Worst-case	0	0	0	0	0	0	0	7
SD	0.89	1.12	1.21	1.69	1.07	1.15	1.18	1.47
Test instance A:								
Mean	−1.78	−1.85	−1.97	−3.25	−1.85	−1.93	−2.25	8.85
Worst-case	0	0	0	0	0	0	0	11
SD	0.76	0.89	0.91	1.49	0.76	0.93	0.98	1.28
Test instance B:								
Mean	−2.98	−3.12	−3.25	−4.96	−3.52	−3.75	−3.98	3.85
Worst-case	0	0	0	0	0	0	0	5
SD	1.16	1.21	1.47	2.01	1.28	1.55	1.62	1.85

Table 4.4: The statistical performance comparison of the different scheduling policies in terms of mean, worst-case, and SD for the clinical and surgical access measures with respect to the maximum wait target (in days) for the case study, as well as the test instances A and B of the CAS problem. The numbers are calculated over all patients whose referrals are received in the 5-day horizon.

Table 4.4 demonstrates empirical results of comparing various policies in terms of clinical and surgical access measures with respect to the target. We summarize the system performance by averaging across all patient classes. We observe that the stochastic-robust, stochastic, and deterministic policies all yield negative clinical and surgical access delay measures. This is because the three policies are able to grant the predefined priority-based access targets to all patients. However, the current policy performs quite differently. While it performs better than the other three policies in terms of providing early access to a clinic consultation appointment, it often fails to provide the crucial surgical appointment within the safe time window, thus compromising health outcomes especially for acute patients.

We next graph the daily mean of surgical access measures with respect to the maximum wait target by the day of referral arrival over 10 days for the case study in Figure 4.7. Again, we summarize the system performance by averaging over all patient classes.

From Figure 4.7, it can be observed that the current policy consistently yields significantly higher wait time to surgery compared to all other polices. This implies a major drawback of the current policy that patients often need to wait a long time to receive a surgical appointment, which can deteriorate their condition. The other three policies, however, uniformly provide on-time (often early) access to surgical procedure.

In conclusion, our coordinated stochastic-robust policy obtains the lowest overtime with

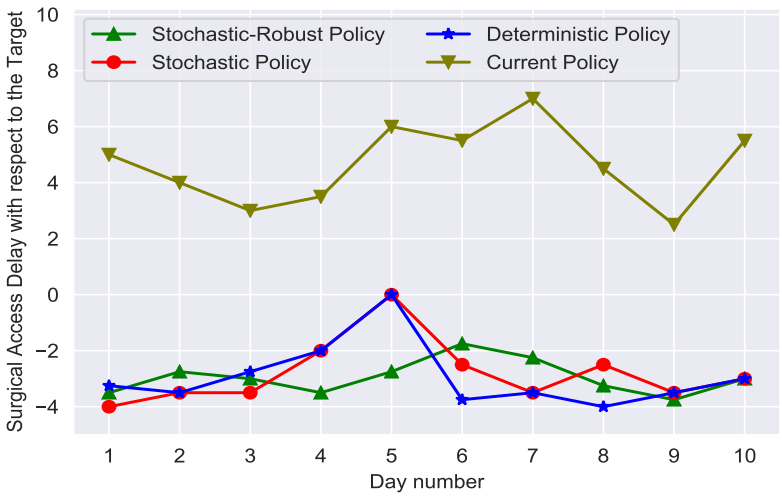


Figure 4.7: The comparison of the surgical access measure with respect to the maximum wait target (averaged across patient classes) by the day of referral arrival obtained by the stochastic-robust, stochastic, deterministic and current policies over the 10-day horizon by the RHP.

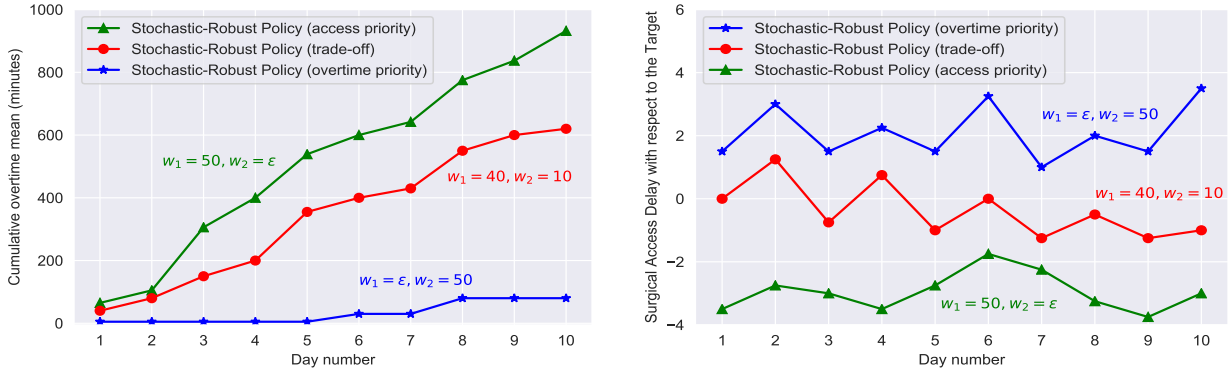
the smallest variability while respecting both clinical and surgical access limits (through imposing the clinical and surgical access constraints (4.4)-(4.9)) so that it finds safe clinical and (if needed) surgical appointments within the target window.

4.6.3 Access Delay versus Overtime Trade-off Analysis

As emphasized in §4.3, the IMSDRO-APRX model ensures *patient-centered care* by providing 100% service level in terms of granting access targets to all patients while optimizing the overtime. However, in Appendix C, we formulate an alternative IMSDRO-APRX model that establishes a trade-off between meeting access delay targets and incurring overtime. This is a multi-objective optimization model, which minimizes (i) the expected penalty due to not meeting clinical and surgical access delay targets (weighted by w_1), and (ii) the maximum expected penalty due to incurring overtime (weighted by w_2). Here, we investigate this balance through implementing the RHP by the stochastic-robust policy obtained from solving the alternative IMSDRO-APRX model.

We calculate the cumulative overtime mean of each surgeon, and the mean of surgical access measures with respect to the maximum wait target by each day. We consider three possible scenarios: (i) the "stochastic-robust policy (access priority)", which puts a large penalty on not meeting the access delay targets ($w_1 = 50, w_2 = \epsilon$), (ii) the "stochastic-robust policy (overtime priority)", which puts a large penalty on the overtime incurred ($w_1 = \epsilon, w_2 = 50$), and (iii) the "stochastic-robust policy (trade-off)", which aims at striking

a balance between these two objectives ($w_1 = 40, w_2 = 10$). Figure 4.8 demonstrates the results for the case study.



(a) Cumulative overtime mean of each surgeon.

(b) Surgical access delay mean with respect to the target.

Figure 4.8: The illustration of trade-off between not meeting access delay targets and incurring overtime for the case study. The cumulative surgeon overtime mean and surgical access delay mean with respect to the target are obtained by three different stochastic-robust policies over a 10-day roll-out window by the RHP for the case study.

As seen in Figure 4.8, while the stochastic-robust policy (access priority) has the highest cumulative overtime mean per surgeon, it provides the patients with the fastest surgical access compared to the other two stochastic-robust polices.

It is also worth noting that the stochastic-robust policy (access priority) yields a surgical access measure mean of -2.1 days with respect to the maximum wait target (i.e., 2.1 days earlier than the deadline). This is about 55% better than the current policy, which yields a surgical access measure mean of 4.65 days (see Table 4.4). This occurs because unlike the current policy, which is a heuristic, the stochastic-robust policy (access priority) solves an optimization model to make appointment decisions.

These observations suggest that the alternative IMSDRO-APRX model proposed in Appendix C is a valuable model that allows decision makers to establish a trade-off between access tardiness and surgeon overtime.

4.6.4 Sensitivity Analysis Results

In this section, we conduct in-sample stability and sensitivity analyses on the importance of care coordination, the probability of needing a surgical procedure, number of days in an arrival horizon, and number of intervals for segment points for the support of surgery duration distribution. Moreover, we compare the single versus multi-cut versions of the constraint generation algorithm.

In-sample Stability Analysis. There are two essential criteria, including (i) in-sample and (ii) out-of-sample stability, to evaluate the efficiency of a scenario tree construction method ([96]). In §4.6.2, we show that the out-of-sample errors are 3.10% (case study), 1.93% (test instance A) and 2.99% (test instance B). In here, we evaluate the in-sample stability. If $|J|$ scenario trees ξ_j, $j \in J$ are generated by using our scenario tree construction method (see Appendix B), and we then solve the IMSDRO-APRX model for each of these scenario trees to calculate the optimal decision vector x_j^* with objective function $f(x_j^*, \xi_j)$ for scenario tree $j \in J$, then *in-sample stability* implies $f(x_j^*, \xi_j) \approx f(x_u^*, \xi_u)$, $\forall j, u \in J$. To evaluate the in-sample stability, we generate different scenario fans with 100 scenarios for the number of appointment requests by the Latin Hypercube Sampling method. Then, a forward scenario construction approach is applied to construct a scenario tree by using different values for the parameter $\boldsymbol{\zeta}_{rel}$ (see Appendix B). Recall that $\boldsymbol{\zeta}_{rel}$ represents a reduction scale of the scenario tree compared with the scenario fan. For each instance with different scenario trees and various values of $\boldsymbol{\zeta}_{rel}$, the *in-sample stability error* is calculated by:

$$\text{In-sample Stability Error} = \frac{\text{Max of objective values} - \text{Min of objective values}}{\text{Average of objective values}} \times 100\%.$$

The number of scenarios decreases as the value of $\boldsymbol{\zeta}_{rel}$ increases.

Test Instance	$\boldsymbol{\zeta}_{rel} = 0.8$			$\boldsymbol{\zeta}_{rel} = 0.7$		
	# of Scenarios	Objective fun.	in-sample error	# of Scenarios	Objective fun.	in-sample error
The case study	7	4,625	4.64%	13	4,545	3.29%
	8	4,561		14	4,451	
	10	4,474		15	4,478	
	11	4,687		17	4,578	
Test instance A	9	6,625	3.03%	17	6,488	2.21%
	10	6,737		18	6,536	
	11	6,536		20	6,421	
	13	6,585		22	6,485	
Test instance B	10	2,265	1.99%	17	2,254	1.85%
	11	2,235		19	2,289	
	13	2,280		20	2,247	
	14	2,280		21	2,265	

Table 4.5: The in-sample stability analysis: Illustration of the empirical results of in-sample stability analysis for the scenario tree construction approach used for the case study, and test instances A and B.

Table 4.5 shows the empirical results of the in-sample stability analysis for the case study, and test instances A and B. The difference between the objective function values with different scenario trees is smaller (i.e., smaller in-sample error) when $\boldsymbol{\zeta}_{rel}$ is set to 0.7. More importantly, the lack of any substantial difference between the optimal objective function values indicates a very good in-sample stability of our scenario tree construction approach.

Importance of Care Coordination. In the case study, about one in three appointment

requests will end up requesting a surgical procedure after the clinic consultation appointment. Therefore, the expected probability of needing surgery is 0.33, based on which we draw the sample paths. To assess the importance of care coordination, we implement the RHP for the case study under two scenarios: (i) considering the surgery need, and (ii) ignoring the possibility of surgery need for patients. We then investigate how the stochastic-robust policy performs under these two scenarios by calculating the mean and variability of the cumulative overtime values over 10 days for one surgeon. Figure 4.9 illustrates the results for the comparison of considering versus ignoring the surgery need.

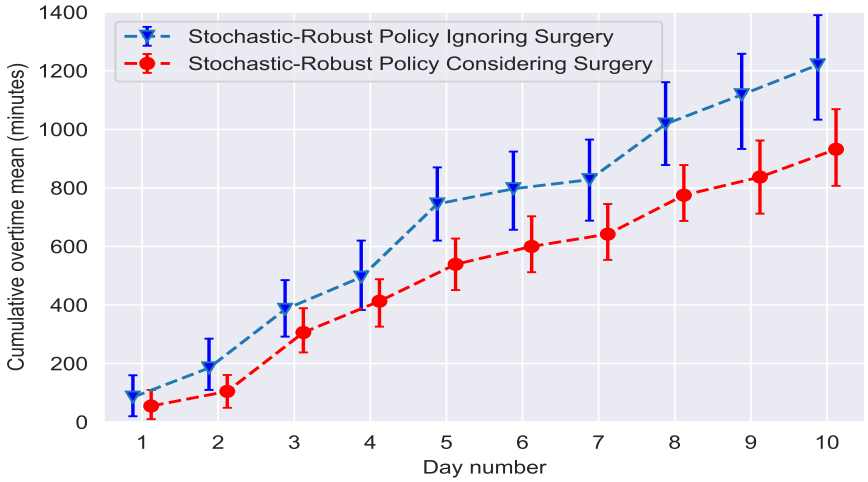
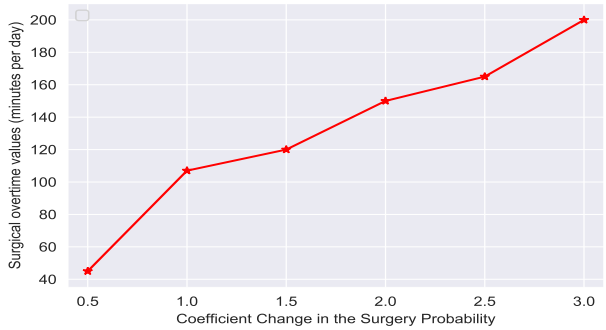


Figure 4.9: The importance of care coordination: a comparison of cumulative overtime means for the stochastic-robust policy in the case study under ignoring versus considering the surgery need over 10 days.

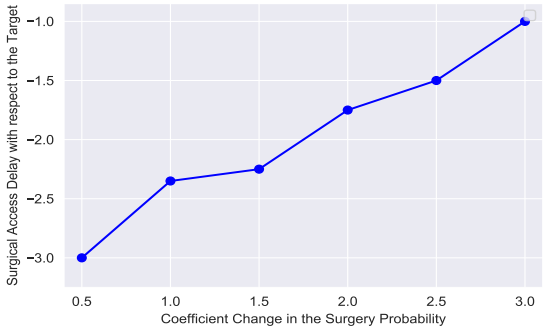
In Figure 4.9, the blue curve only considers the already booked surgeries over the next periods; it ignores the likelihood of future possible surgeries. On the other hand, the red curve does take the probability of future surgeries into account when making appointment scheduling decisions. As seen in Figure 4.9, the stochastic-robust policy performs significantly better (i.e., less overtime) when the probability of future surgeries are taken into account. In particular, the stochastic-robust policy considering surgery probabilities yields a daily overtime mean of 90 minutes for a surgeon, which is about 26% better (less overtime) than the stochastic-robust policy ignoring the probability of surgeries with a daily overtime mean of 122 minutes for a surgeon. It is worth noting that, in our case study, we defined the "regular time" to be 6 hours per day for each surgeon. This is, of course, a conservative way of defining the regular versus overtime. However, given that the objective function minimizes overtime, defining a conservative regular time allows the model to be mindful of scheduling appointments beyond 6 hours per day. Therefore, a mean overtime of 90 minutes observed

in Figure 4.9 implies that the surgeons will work, on average, 7.5 hours per day under the stochastic-robust policy that considers future surgery needs. To sum up, we demonstrate that the idea of care coordination can help to achieve less overtime by considering the uncertainty around future surgery needs.

Sensitivity Analysis on the Surgery Probability. In our approach, we model the need for a surgical procedure through a Bernoulli random variable. This assumption is made by others as well (see e.g., [153] and [98]). We evaluate this modeling choice by a sensitivity analysis on the surgery probability with respect to both surgical overtimes and surgical access time delays. Figure 4.10 illustrates how the surgical overtime and average surgical access time delay change as the surgery probability alters. We observe that as the surgery probability increases, we require more surgical overtime while the surgical access delay increases.



(a) Surgical overtime versus the surgery prob.



(b) Surgical access delay versus the surgery prob.

Figure 4.10: The sensitivity analysis around the surgery probability with respect to (a) surgical overtime, and (b) surgical access measure (negative values indicate earliness, i.e., the model grants access to surgery within the maximum wait time target for each patient.)

Importance of Number of Days in the Arrival Horizon. In the case study, we consider a 5-day arrival horizon. In other words, we consider the uncertainty around the number of appointment requests and the surgery durations of the next 5 days when making appointment scheduling decisions. In this analysis, we investigate how the number of days considered in the arrival horizon affects the quality of the stochastic-robust policy. Figure 4.11 demonstrates the performance of the stochastic-robust policy by computing the cumulative overtime mean for a surgeon obtained from solving the IMSDRO-APRX model by using the RHP for 10 days in our case study. We consider three different arrival horizons with $T = |\mathcal{T}| = 3$, 5 and 7 days.

From Figure 4.11, we observe that as the number of days in the arrival horizon increases, the performance of the stochastic-robust policy gradually improves (i.e., less overtime is

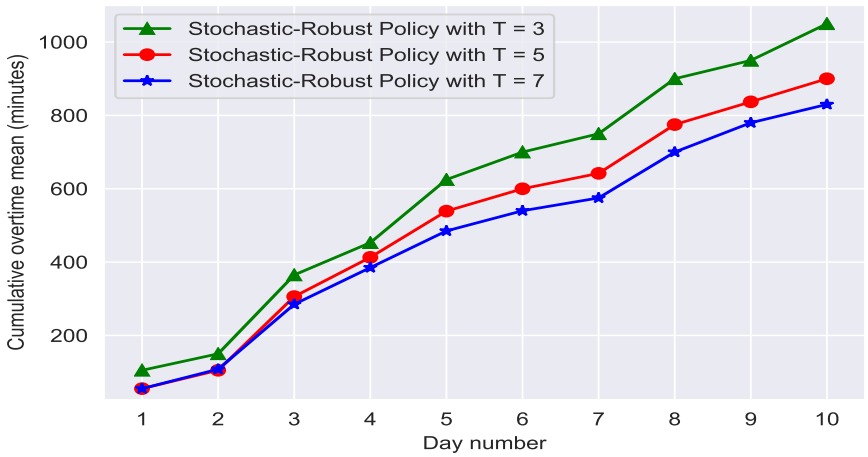


Figure 4.11: Importance of the number of days for the arrival horizon $\mathcal{T}$: the performance of the stochastic-robust policies with $T = 3$, 5, and 7 days in terms of cumulative overtime mean per surgeon over 10 days for the case study obtained by solving the IMSDRO-APRX model by using the RHP.

incurred) as we roll forward in the arrival horizon. The longer the arrival horizon, the less myopic the policy. This is because longer-term uncertainty about the number of patient appointment requests and surgery duration is taken into account when the stochastic-robust policy makes the clinical and surgical decisions for patients. As seen in Figure 4.11, reducing the length of arrival horizon from 5 to 3 days increases the overtime mean per surgeon by about 15 minutes on day 10. However, increasing the length of arrival horizon from 5 to 7 days only reduces the overtime mean per surgeon by 6 minutes on day 10 (the cumulative overtime means are 1050, 900 and 840 minutes by day 10, for $T = 3, 5$ and 7, respectively, which are equivalent to 105, 90 and 84 minutes per surgeon per day). This suggests that while in general including *additional days* in the arrival horizon is helpful, increasing the horizon from 5 to 7 days has little benefit and may not worth the additional computational burden.

Analysis of Support Discretization for the Surgery Duration. In §4.3, we reformulated the objective function (4.24) into a piece-wise linear function with H equal intervals through discretizing the support set $[d^{LB}_{\gamma,k}, d^{UB}_{\gamma,k}]$ of the surgery distribution $d_{\gamma,k}$ for each pair of class $\gamma \in \Gamma$ and surgeon $k \in \mathcal{K}$ into $H+1$ segment points $\Upsilon_{\gamma,k} = \{\tilde{d}_{\gamma,k}(i)\}_{i=0}^{H}$. Here, we provide sensitivity analysis results on varying the number of segment points and investigate the trade-off between the solution quality and the computational time of solving the IMSDRO-APRX model. Intuitively, when the number of segment points H increases, it results in achieving a more precise approximation for the objective function, but with a longer computational time.

Test instance	# of segment points	Objective fun.	# of Iterations	CPU time
The case study	5	4,625	7	1,014
	10	4,451	9	1,345
	20	4,375	13	2,116
Test instance A	5	6,757	6	1,546
	10	6,421	10	2,005
	20	6,235	15	2,643
Test instance B	5	2,389	5	1,189
	10	2,265	8	2,115
	20	2,218	16	2,954

Table 4.6: The sensitivity analysis on the number of segment points and how it impacts the objective function value, computational time, and the number of iterations for the case study and the test instances A and B.

Table 4.6 reports the objective function value, computational time (in seconds), and the number of iterations required for solving the IMSDRO-APRX model for the case study and test instances A and B with various number of segment points for the surgery duration. As we increase the number of segment points for the surgery duration (so the support of the surgery duration is approximated more accurately), we obtain a more precise approximation of the objective function with a larger number of iterations. It also requires a longer computational time; however, note that it grows slower than linearly. As we can see in Table 4.6, by doubling the number of segment points (i.e., increasing it from 5 to 10 and from 10 to 20), the objective function value alters by less than 5% in the case study. Results are similar for test instances A and B. Table 4.6 demonstrates that our choice of using 10 segment points as the default in the case study is appropriate.

Comparison of Single versus Multi-Scenario Cuts. In §4.4, we developed a constraint generation algorithm with *multi-scenario cuts* for solving the IMSDRO-APRX model, in which the cut-generating problem (4.26a)-(4.26f) obtains at most one scenario cut per scenario and passes it back to the RMP (4.29a)-(4.29d). Similar to the L-shaped decomposition methods, our algorithm can have two versions: (i) a multi-cut version in which multiple cuts are added to the RMP (at most one cut per scenario); and (ii) a single cut version in which one aggregated cut is added to the RMP.

In this part, we introduce the single cut version of the proposed constraint generation algorithm, which is similar to Algorithm 4, except that it passes back one aggregated cut in

the form of (4.30b) to the following RMP (4.30a)-(4.30d) at each iteration:

$$\bar{Z}^{RMP} = \min_{\boldsymbol{x},\boldsymbol{y},\widehat{\boldsymbol{y}},\boldsymbol{q},\boldsymbol{\delta},\boldsymbol{\alpha},\boldsymbol{\beta}} \sum_{s\in\mathcal{S}} \pi_s \left\{ \sum_{k\in\mathcal{K}} \Big(\sum_{m\in\mathcal{L}} q^k_{m,s} + \sum_{\gamma\in\Gamma} \Big(\mu_{\gamma,k}\ \alpha^k_{\gamma,s} + (\mu^2_{\gamma,k} + \sigma^2_{\gamma,k})\ \beta^k_{\gamma,s} \Big) \Big) \right\} + \Pi \tag{4.30a}$$

$$\text{s.t.} \quad \Pi \geq \sum_{s\in\mathcal{S}} \pi_s\ \chi_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s}; \boldsymbol{\tau_s^{(r)}}, \boldsymbol{\eta_s^{(r)}}, \boldsymbol{\lambda_s^{(r)}}), \quad \forall r = 1, \cdots, R-1 \tag{4.30b}$$

$$(\boldsymbol{x_s},\ \boldsymbol{y_s}, \widehat{\boldsymbol{y}},\ \boldsymbol{q_s}) \in \mathcal{R}_s, \quad \forall s \in \mathcal{S} \tag{4.30c}$$

$$\delta_s \in \mathbb{R},\ \boldsymbol{\alpha_s},\ \boldsymbol{\beta_s} \in \mathbb{R}^{|\Gamma|\times|\mathcal{K}|}, \quad \forall s \in \mathcal{S}. \tag{4.30d}$$

We next compare the empirical performance of single versus multi-scenario cut versions of the proposed constraint generation algorithm. To this aim, we generate 10 random instances of the CAS problem with different number of surgeons, number of patient classes, and number of scenarios. Each instance is specified by a tuple ($|\mathcal{K}|$,$|\Gamma|$,$|\mathcal{T}|$,$|\mathcal{S}|$) denoting the number of surgeons, patient classes, days in the arrival arrival, and scenarios for patient appointment requests, respectively. We then solve the IMSDRO-APRX model for each of these instances by using both multi-cut and single-cut versions of the constraint generation algorithm (Algorithm 4). Table 4.7 indicates the number of iterations ("# of Iterations") for them until the algorithm converges to the optimal policy as well as the CPU time in seconds for each instance. Note that the instance numbers 5, 8 and 9 refer to the test instance A, the case study and the test instance B, respectively.

Instance Number	($\|\mathcal{K}\|$, $\|\Gamma\|$, $\|\mathcal{T}\|$, $\|\mathcal{S}\|$)	multi cuts		single cut	
		# of Iterations	CPU time	# of Iterations	CPU time
1	(4, 12, 4, 8)	5	654	9	1,177
2	(4, 12, 5, 10)	7	1,021	12	1,897
3	(4, 12, 5, 15)	8	1,254	18	2,257
4	(4, 24, 5, 15)	6	1,578	13	2,957
5	(4, 24, 5, 20)	10	2,005	20	3,609
6	(8, 12, 5, 8)	4	755	13	1,177
7	(8, 24, 5, 10)	6	1,176	13	2,215
8	(8, 24, 5, 14)	9	1,345	15	2,421
9	(10, 24, 5, 10)	8	2,115	16	3,484
10	(10, 24, 5, 15)	7	2,321	13	3,752

Table 4.7: The comparison of the multi-cut and single-cut versions of the constraint generation Algorithm 4 for different instances of the CAS problem in terms of the number of iterations and CPU time in seconds.

Table 4.7 demonstrates that we are able to solve a range of suitable instances for the IMSDRO-APRX model within a reasonable number of iterations. We also observe that since the multi-cut version offers more information about the feasible region, we require fewer

number of iterations compared to the single-cut version. The average number of iterations for the multi-cut version was 6, while it was 14 for the single-cut version. Moreover, the single-cut version takes more CPU time compared to the multi-cut version.

4.7 Practical Implications and Insights

In §4.6, we demonstrated the application of our IMSDRO approach to coordinate clinic and surgery visits in a highly specialized surgical unit. We showed that our model can provide access to surgery within a safe time frame, especially for acute patients who will most suffer from a long wait time, while minimizing the overtime. We summarize the insights and practical implications below.

First and foremost, surgical divisions that offer surgical procedures after a clinic consultation appointment should consider leveraging optimization algorithms that coordinate the clinic and surgery appointments when scheduling new appointments. Simple heuristic scheduling protocols, such as scheduling new appointment requests with the surgeon who has the earliest availability (i.e., the "current policy" in our case study), often result in prolonged wait times for patients with acute conditions like cancer. The lengthy wait times to receive a surgical procedure may result in adverse events and poor patient outcomes. In contrast, through minimizing the overtime, our proposed coordinated stochastic-robust policy achieved both clinical and surgical access targets, which were stratified into five classes based on patients' acuity level. Simple heuristics, including the one used by our partner hospital, may allocate clinic and surgery appointment dates for a patient several days beyond the acceptable wait time target windows. This is not clinically safe for patients and may lead to additional complications. The success of our algorithmic, optimization-based method indicated that it is not always effective to offer the earliest available appointment slot to a new patient as commonly done in current policy. If the patient has an acute condition, consideration of the likelihood of surgery and availability of providers is key to ensure timely access to surgery.

Moreover, even though our model considered overtime to meet the priority-based access to care targets, our empirical results showed that the mean overtime per surgeon is around 90 minutes. It should be noted that we defined the regular time for surgeons as 6 hours per day in the case study. Thus, the mean overtime of 90 minutes per day means that a typical day for surgeons last 7.5 hours, on average. We also demonstrated that our stochastic-robust policy achieves the lowest overtime and the smallest variability among the four policy we investigated, while respecting both clinical and surgical access limits. The average workday of 7.5 hours, together with granting access targets of the stochastic-robust policy, confirm

that the appointment scheduling plans obtained from our IMSDRO approach are feasible and implementable in practice. Our optimization models provide generality over a broader range of operation systems and parameters than most heuristics, which do not readily extend to new settings. Our analytic approach allows the administration to modify the parameters of the system to find an acceptable optimal policy. For instance, if the amount of overtime suggested by our model is not desirable, the administration can relax the priority-based access targets to reduce the required overtime. If new surgeons are hired or new procedures are offered, the model can be easily extended to accommodate the new conditions.

Modeling care coordination in our coordinated stochastic-robust policy results in better utilization of scarce resources, including surgeon time and operating rooms. We saw in Figure 4.9 that the policy that takes uncertain future surgeries into consideration outperforms the policy that ignores the uncertainty of the need for surgery (26% less overtime). Moreover, we demonstrated in multiple ways that the stochastic-robust policy achieves much lower variability in surgeon overtime and patient access time compared to alternative policies. This is extremely important in healthcare setting since avoiding extreme scenarios and achieving a reliable performance will allow the hospital management to better control patient flow and manage their resources and processes.

Our research promotes patient-centered care by stratifying patients into different priority classes based on what is known about the patient at the time the patient referral is received (e.g., the indication of disease), which are then translated into appropriate and safe maximum wait time targets. Surgical divisions should also take the uncertainty in appointment request arrival, surgical demand, and surgery durations into account when scheduling clinic consultation and surgery appointments. Our models provide a creative way to do so using data that are commonly available in the patient's electronic health records and the clinic's datasets, and do not rely on assumptions on the probability distribution of surgeries. Further, the proposed data-driven rolling horizon procedure introduces an innovative way of making use of the latest data that is revealed as time progresses, and adjusting the decisions in practice for stochastic optimization problems.

Furthermore, we provided two optimization models derived by our IMSDRO approach. The focus of the first (main) one was on ensuring patient-centered care by providing 100% service level in terms of meeting access delay targets to all patients while minimizing the surgeon overtime. This was the model that our partner hospital preferred. The second optimization model considered two competing objectives, namely, meeting access delay targets and incurring overtime. As illustrated in Figure 4.8, this model allows decision makers to establish a trade-off between providing timely access to care to patients and asking surgeons to work overtime hours.

Our coordinated stochastic-robust policy improves the surgical access times by about 160%, on average, compared to the current policy used by our partner hospital (see Table 4.4). Intuitively, this is because our methods take into account the wait time target windows as well as various inherent sources of uncertainty, including the number of appointment requests, probability of surgery need, and surgery duration while coordinating clinic and surgery appointments. Also, the current policy ignores the valuable indication of disease that is available in the patient's electronic health records when a new appointment request is received. Unlike the current policy that operates based on a first-come first-serve idea, intuitively our model often defers the surgery of low-priority patients in order to preserve the near future capacity to serve high-priority patients that may arrive later. This approach helps meet the desired service level with minimum overtime.

Although our work is motivated by healthcare, our models and insights can also be applied to other service industries. For example, there are IT service companies with contracts that incorporate service level agreements to maintain the IT infrastructure for many client companies. [136] describe client-specific priority levels specified in these agreements as well as deadlines for the completion of a client's service request in response to a failure. To efficiently utilize human resources and meet the various service level agreements, the IT company must schedule a date to work on an arriving service request. It may be possible to resolve the service request in that first visit; however, the first service may be used as a triage phase to identify when the problem is time-consuming and requires a larger effort to be scheduled for a later date so that this job does not delay other higher priority requests. This operational approach is analogous to the problem here, which has distinct experts providing one or two phases of service with a goal of completing the work within the deadline by effectively scheduling service activities. Because there are many different types of jobs/issues that the company addresses using their experts (or teams), estimating the full distribution of service time for each job is not practical. Thus, a robust approach is beneficial. Note that in many of such problems, the job *must* be completed within the time window specified in the contract even if some overtime is required. In this case, a model similar to the one considered in our paper that guarantees pre-defined service levels is appropriate. In some cases it might be possible to complete the job with some delay to reduce the amount of overtime needed. If this is the case, a revised model formulation such as the one we provide in the appendix can strike a balance between job completion delay and personnel overtime.

4.8 Conclusion, Limitations, and Future Research

In this paper, we studied a new class of appointment scheduling problems called the "coordinated clinic and surgery appointment scheduling" in which patients are stratified into patient classes, with limits on the allowable access delay from request to appointment dates. We introduced the concept of care coordination in the sense of setting appointments for pairs of sequential clinic and (if needed) surgery visits that together achieve timely access to care. Methodologically speaking, our integrated multi-stage stochastic and distributionally robust optimization (IMSDRO) is the first optimization approach that can jointly incorporate different types of uncertainty in the number of patient appointment requests by a scenario tree, and in surgery durations by a moment-based ambiguity set for distributional robustness. Using the special structure of the CAS problem, we proposed a constraint generation algorithm for efficiently solving this problem. We then developed a new data-driven rolling horizon procedure to implement the decisions made by the IMSDRO approach in practice. This allows healthcare practitioners to make efficient use of data that is obtained as time unfolds, and so adjust their decisions in a rolling horizon framework. In a sense, our methods/models can be applied in an online (or real-time) fashion. We tested the validity of our models/algorithms in a case study of scheduling clinic consultation and surgery appointments, and demonstrated that a significant improvement could be achieved if our partner hospital were to switch from the current heuristic scheduling protocol to our proposed policies. We provide several practical insights from our empirical analysis as well.

This study has a few limitations. In our models, we do not consider patient no-shows and cancellations as well as the potential seasonality in demand as they rarely happen in our highly-specialized partner surgical suites. Patient preferences are also not part of our models and algorithms. Clearly, in many health care environments, the patient can prioritize the selection of the provider with whom they feel most comfortable. Our scope is; however, limited to the important class of environments in which the patients typically accept the provider offering the earliest access. Moreover, the allocation of resources including operating rooms to surgeons is not the main focus in our paper. Finally, given that a tractable system state can be defined, approximate or robust dynamic programming approaches may be used to solve the CAS problem. These ideas could be promising future research directions in the area of appointment scheduling.

4.9 Appendix

4.9.1 Appendix A: Technical Proofs for the Analytical Results.

A.1. Proof of Proposition 11

Proof. When the probability measure $P \in \Phi(\boldsymbol{\mu}, \boldsymbol{\sigma}, \Theta)$ defined by the polyhedral support set (4.18), we can explicitly rewrite the moment problem (4.21) under each individual scenario $s \in \mathcal{S}$ as follows:

$$\max_{P \in \Phi(\boldsymbol{\mu}, \boldsymbol{\sigma}, \Theta)} \mathbb{E}_P \Big[f_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{d}) \Big] \;=\; \max_{P} \left\{ \int_{\Theta} f_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{d}) \; dP(d) \right\} \tag{4.31}$$
$$\text{s.t.} \quad \text{constraints } (4.19a) - (4.19c),$$

as a linear program maximizing over all plausible distributions P in the set $\Phi(\boldsymbol{\mu}, \boldsymbol{\sigma}, \Theta)$, and with the expectation of the function $f_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{d})$ taken over this distribution as the objective function. Since all $\boldsymbol{y_s}$, $\widehat{\boldsymbol{y}}$, and $\boldsymbol{d}$ vectors are input parameters to the moment problem (4.21), and it contains the continuous decision variable P and linear constraints (4.19a)-(4.19c), we can take the dual of linear program (4.31) by associating the dual variable vectors $\delta_s \in \mathbb{R}$, $\boldsymbol{\alpha_s} \in \mathbb{R}^{|\Gamma| \times |\mathcal{K}|}$ and $\boldsymbol{\beta_s} \in \mathbb{R}^{|\Gamma| \times |\mathcal{K}|}$ with the constraints (4.19a)-(4.19c), respectively, for each individual scenario $s \in \mathcal{S}$. Thus, its dual is a *semi-infinite linear* program as follows:

$$\min_{\boldsymbol{\delta}, \boldsymbol{\alpha}, \boldsymbol{\beta}} \quad \delta_s + \sum_{\gamma \in \Gamma} \sum_{k \in \mathcal{K}} \mu_{\gamma,k} \; \alpha^k_{\gamma,s} + \sum_{\gamma \in \Gamma} \sum_{k \in \mathcal{K}} (\mu^2_{\gamma} + \sigma^2_{\gamma,k}) \; \beta^k_{\gamma,s} \tag{4.32a}$$
$$\text{s.t.} \quad \delta_s + \sum_{\gamma \in \Gamma} \sum_{k \in \mathcal{K}} d_{\gamma,k} \; \alpha^k_{\gamma,s} + \sum_{\gamma \in \Gamma} \sum_{k \in \mathcal{K}} d^2_{\gamma,k} \; \beta^k_{\gamma,s} \;\geq\; f_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{d}), \quad \forall \boldsymbol{d} \in \Theta. \tag{4.32b}$$

Using the strong duality theorem, we next substitute the inner maximization moment problem (4.21) with (4.32a)-(4.32b) in the min-max IMSDRO model (4.20a)-(4.20b), and merge the minimization objective (4.32a) with the minimization objective in the IMSDRO model (4.20a)-(4.20b) to obtain a reformulation of the min-max IMSDRO model (4.20a)-

(4.20b) under the ambiguity set $\Phi(\boldsymbol{\mu}, \boldsymbol{\sigma}, \Theta)$ as

$$\min_{\boldsymbol{x},\boldsymbol{y},\widehat{\boldsymbol{y}},\boldsymbol{q},\boldsymbol{\delta},\boldsymbol{\alpha},\boldsymbol{\beta}} \quad \sum_{s\in\mathcal{S}} \pi_s \left\{ \sum_{k\in\mathcal{K}}\sum_{m\in\mathcal{L}} q^k_{m,s} + \delta_s + \sum_{\gamma\in\Gamma}\sum_{k\in\mathcal{K}} \mu_{\gamma,k}\ \alpha^k_{\gamma,s} + \sum_{\gamma\in\Gamma}\sum_{k\in\mathcal{K}} (\mu^2_{\gamma,k} + \sigma^2_{\gamma,k})\ \beta^k_{\gamma,s} \right\} \tag{4.33a}$$

$$\text{s.t.} \quad \delta_s + \sum_{\gamma\in\Gamma}\sum_{k\in\mathcal{K}} d_{\gamma,k}\ \alpha^k_{\gamma,s} + \sum_{\gamma\in\Gamma}\sum_{k\in\mathcal{K}} d^2_{\gamma,k}\ \beta^k_{\gamma,s} \quad \geq \quad f_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{d}), \quad \forall \boldsymbol{d} \in \Theta,\ s \in \mathcal{S} \tag{4.33b}$$

$$(\boldsymbol{x_s},\ \boldsymbol{y_s}, \widehat{\boldsymbol{y}},\ \boldsymbol{q_s}) \ \in \ \mathcal{R}_s, \quad \forall s \in \mathcal{S} \tag{4.33c}$$

$$\delta_s \ \in \ \mathbb{R},\ \boldsymbol{\alpha_s},\ \boldsymbol{\beta_s} \ \in \ \mathbb{R}^{|\Gamma|\times|\mathcal{K}|}, \quad \forall s \in \mathcal{S}. \tag{4.33d}$$

However, the reformulation (4.33a)-(4.33c) of the IMSDRO model is still *intractable* since constraint (4.33b) is a semi-infinite constraint, meaning that it should be satisfied for any possible realization of d from the polyhedral support set Θ in (4.18). To obtain a tractable reformulation, since constraint (4.33b) should be satisfied for all realization of $\boldsymbol{d} \in \Theta$, it should be satisfied for the worst-case possible value of $\boldsymbol{d} \in \Theta$. Hence, we move all the terms which contain $\boldsymbol{d}$ to the right-hand side of constraint (4.33b) to obtain the minimization reformulation (4.22a)-(4.22d), which completes the proof. □

A.2. Proof of Proposition 12

Proof. For each individual scenario $s \in \mathcal{S}$, we can derive the following simple equivalent linear program (LP) for the function $f_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{d})$ using the surgical overtime definition for surgeons as follows:

$$f_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{d}) = \min_{\boldsymbol{o}} \quad \sum_{k\in\mathcal{K}}\sum_{n\in\mathcal{L}} o^k_{n,s} \tag{4.34a}$$

$$\text{s.t.} o^k_{n,s} \geq \sum_{\gamma\in\Gamma} d_{\gamma,k}\Big(\sum_{t\in\mathcal{U}}\sum_{m\in\mathcal{T}\setminus\{t_0\}} \widetilde{y}^{k,n}_{\gamma,t,m} + \sum_{t\in\mathcal{T}}\sum_{m\in\mathcal{T}\setminus\{t_0\}} y^{k,n}_{\gamma,t,m,s} + \sum_{t\in\mathcal{T}\cup\{t_0\}} \widehat{y}^{k,n}_{\gamma,t,t_0}\Big) - V^k_n,$$

$$\forall k \in \mathcal{K}, n \in \mathcal{L} \tag{4.34b}$$

$$o^k_{n,s} \geq 0, \quad \forall k \in \mathcal{K}, n \in \mathcal{L}. \tag{4.34c}$$

Next, we take the dual formulation of the LP (4.34a)-(4.34c) by associating dual variables

$\lambda^k_{n,s} \in \mathbb{R}_+$ to the constraints (4.34b) as follows for each individual scenario $s \in \mathcal{S}$:

$$f_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{d}) = \max_{\lambda_s} \sum_{k\in\mathcal{K}} \sum_{n\in\mathcal{L}} \Big\{ \sum_{\gamma\in\Gamma} d_{\gamma,k} \Big(\sum_{t\in\mathcal{U}} \sum_{m\in\mathcal{T}\setminus\{t_0\}} \widetilde{y}^{k,n}_{\gamma,t,m} \tag{4.35a}$$

$$+ \sum_{t\in\mathcal{T}} \sum_{m\in\mathcal{T}\setminus\{t_0\}} y^{k,n}_{\gamma,t,m,s} + \sum_{t\in\mathcal{T}\cup\{t_0\}} \widehat{y}^{k,n}_{\gamma,t,t_0} \Big) - V^k_n \Big\} \lambda^k_{n,s}$$

$$\text{s.t.} \;\; 0 \;\leq\; \lambda^k_{n,s} \;\leq\; 1, \;\; \forall k \in \mathcal{K}, n \in \mathcal{L}. \tag{4.35b}$$

We then substitute the dual problem (4.35a)-(4.35b) into the maximization problem $\Psi_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s})$ on the right hand side of the constraints (4.22b), which results in the following equivalent problem:

$$\Psi_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s}) = \max_{d\in\Theta} \Big\{ \max_{\lambda_s\in\Lambda_s} \sum_{k\in\mathcal{K}} \sum_{n\in\mathcal{L}} \Big\{ \sum_{\gamma\in\Gamma} d_{\gamma,k} \Big(\sum_{t\in\mathcal{U}} \sum_{m\in\mathcal{T}\setminus\{t_0\}} \widetilde{y}^{k,n}_{\gamma,t,m} + \sum_{t\in\mathcal{T}} \sum_{m\in\mathcal{T}\setminus\{t_0\}} y^{k,n}_{\gamma,t,m,s} + \tag{4.36}$$

$$\sum_{t\in\mathcal{T}\cup\{t_0\}} \widehat{y}^{k,n}_{\gamma,t,t_0} \Big) - V^k_n \Big\} \cdot \lambda^k_{n,s} - \sum_{\gamma\in\Gamma} \sum_{k\in\mathcal{K}} d_{\gamma,k}\, \alpha^k_{\gamma,s} - \sum_{\gamma\in\Gamma} \sum_{k\in\mathcal{K}} d^2_{\gamma,k}\, \beta^k_{\gamma,s} \Big\}.$$

Swapping the order of maximizations in (4.36) does not affect the optimal solution because the polyhedron-shaped support set Θ of $\boldsymbol{d}$ is a *compact and bounded set*. Thus, as we have a separable structure for the polyhedron-shaped support set Θ of $\boldsymbol{d}$, we can maximize the objective function (4.36) first over $\lambda_s \in \Lambda_s$, and then over $\boldsymbol{d} \in \Theta$, separately as follows:

$$\Psi_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s}) = \max_{\lambda_s\in\Lambda_s} \Big\{ \max_{d\in\Theta} \Big\{ \sum_{\gamma\in\Gamma} \sum_{k\in\mathcal{K}} \Big(\sum_{n\in\mathcal{L}} \Big\{ \sum_{t\in\mathcal{U}} \sum_{m\in\mathcal{T}\setminus\{t_0\}} \widetilde{y}^{k,n}_{\gamma,t,m} + \sum_{t\in\mathcal{T}} \sum_{m\in\mathcal{T}\setminus\{t_0\}} y^{k,n}_{\gamma,t,m,s} \tag{4.37}$$

$$+ \sum_{t\in\mathcal{T}\cup\{t_0\}} \widehat{y}^{k,n}_{\gamma,t,t_0} \Big\} \cdot \lambda^k_{n,s} d_{\gamma,k} - \alpha^k_{\gamma,s} d_{\gamma,k} - \beta^k_{\gamma,s} d^2_{\gamma,k} \Big) \Big\} - \sum_{k\in\mathcal{K}} \sum_{n\in\mathcal{L}} V^k_n \lambda^k_{n,s} \Big\}.$$

Next, given the fact that the polyhedron-shaped support set Θ of $\boldsymbol{d}$ is defined by independent lower and upper bounds in each dimension of $\gamma \in \Gamma$ and $k \in \mathcal{K}$ pair (see (4.16)), the inner maximization problem over $\boldsymbol{d} \in \Theta$ in the problem (4.37) is a separable optimization problem by the patient class $\gamma \in \Gamma$ and surgeon $k \in \mathcal{K}$ indices. We can separate this inner max problem in the problem (4.37) into $|\Gamma| \times |\mathcal{K}|$ maximization problems, each of them is over the interval $d^{LB}_{\gamma,k} \leq d_{\gamma,k} \leq d^{UB}_{\gamma,k}$, and make a summation over the indices $\gamma \in \Gamma$ and $k \in \mathcal{K}$, which results in the optimization problem (4.23) and completes the proof. □

A.3. Proof of Theorem IV.1

Proof. For each pair of patient class $\gamma \in \Gamma$ and surgeon $k \in \mathcal{K}$, we define binary variables $\eta^k_{\gamma,i} \in \{0,1\}$ corresponding to each segment point $\tilde{d}_{\gamma,k}(i)$, $i = 0, \cdots, H$ such that $\eta^k_{\gamma,i} = 1$ if the i^{th} segment point $\tilde{d}_{\gamma,k}(i)$ in the set $\Upsilon_{\gamma,k} = \{\tilde{d}_{\gamma,k}(i)\}_{i=0}^{H}$ yields the maximum value for the problem (4.25) and $\eta^k_{\gamma,i} = 0$ otherwise. Consequently, the approximation problem (4.25) for the given $\boldsymbol{y}, \boldsymbol{\alpha}$, and $\boldsymbol{\beta}$ decisions is equivalent to the following problem for each pair of patient class $\gamma \in \Gamma$ and surgeon $k \in \mathcal{K}$ under scenario $s \in \mathcal{S}$:

$$\max_{\boldsymbol{\eta}} \Bigg\{ \Bigg(\sum_{i=0}^{H} \Bigg(\sum_{n\in\mathcal{L}} \Big\{ \sum_{t\in\mathcal{U}} \sum_{m\in\mathcal{T}\setminus\{t_0\}} \widetilde{y}^{k,n}_{\gamma,t,m} + \sum_{t\in\mathcal{T}} \sum_{m\in\mathcal{T}\setminus\{t_0\}} y^{k,n}_{\gamma,t,m,s} + \sum_{t\in\mathcal{T}\cup\{t_0\}} \widehat{y}^{k,n}_{\gamma,t,t_0} \Big\} \lambda^k_{n,s} \Bigg) \tilde{d}_{\gamma,k}(i) - \alpha^k_{\gamma,s}\, \tilde{d}_{\gamma,k}(i) - \beta^k_{\gamma,s}\, \tilde{d}_{\gamma,k}(i)^2 \Bigg) \eta^k_{\gamma,i} \Bigg\} \tag{4.38a}$$

$$\text{s.t.} \quad \sum_{i=0}^{H} \eta^k_{\gamma,i} \;=\; 1, \tag{4.38b}$$

$$\eta^k_{\gamma,i} \in \{0,1\}, \quad \forall i = 0, \cdots, H. \tag{4.38c}$$

To ensure that exactly one of the segment points $\tilde{d}_{\gamma,k}(i)$, $i \in \{0, \cdots, H\}$ is selected for each pair (γ, k) to maximize the objective function of problem (4.25), we require the constraints (4.38b). Note that each set of segment points $\Upsilon_{\gamma,k} = \{\tilde{d}_{\gamma,k}(i)\}_{i=0}^{H}$ for each pair of class γ and surgeon k is a Specially Ordered Set of Type 1 (SOS1), containing binary variables that sum to one (see constraints (4.38b)). When the segment point $\tilde{d}_{\gamma,k}(i')$ maximizes (4.25), i.e., $\eta^k_{\gamma,i'} = 1$, the objective function of (4.38a) gets the value of $\sum_{i=0}^{H} \Big(\sum_{n\in\mathcal{L}} \Big\{ \sum_{t\in\mathcal{U}} \sum_{m\in\mathcal{T}\setminus\{t_0\}} \widetilde{y}^{k,n}_{\gamma,t,m} + \sum_{t\in\mathcal{T}} \sum_{m\in\mathcal{T}\setminus\{t_0\}} y^{k,n}_{\gamma,t,m,s} + \sum_{t\in\mathcal{T}\cup\{t_0\}} \widehat{y}^{k,n}_{\gamma,t,t_0} \Big\} \lambda^k_{n,s}\, \tilde{d}_{\gamma,k}(i) - \alpha^k_{\gamma,s}\, \tilde{d}_{\gamma,k}(i) - \beta^k_{\gamma,s}\, \tilde{d}_{\gamma,k}(i)^2 \Big)$ and other $\eta^k_{\gamma,i}$ variables are zero for $i \in \{0, \cdots, H\}$, $i \neq i'$.

Furthermore, we see that there is a bi-linear expression $\lambda^k_{n,s}\eta^k_{\gamma,i}$ in the objective function (4.38a) for the given $\boldsymbol{y}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha}$, and $\boldsymbol{\beta}$ decisions. However, we can reformulate these bi-linear terms by adding some *McCormick type inequalities* because $\eta^k_{\gamma,i}$ is a binary variable and we have lower and upper bounds, i.e., $0 \leq \lambda^k_{n,s} \leq 1$, for the variable $\lambda^k_{n,s}$ based on the polyhedron set Λ_s defined in Proposition 12. To do so, we define auxiliary variables $\tau^k_{n,s,\gamma,i}$ such that $\tau^k_{n,s,\gamma,i} = \lambda^k_{n,s}\, \eta^k_{\gamma,i}$ for all $i \in \{0, \cdots, H\}$, and $\gamma \in \Gamma$. To remove theses bi-linear terms in objective function (4.38a), we need to add the McCormick type constraints (4.26c)-(4.26e) [125], and $\tau^k_{n,s,\gamma,i} \geq 0$, which guarantee that $\tau^k_{n,s,\gamma,i} = \lambda^k_{n,s}\, \eta^k_{\gamma,i}$ for all $i \in \{0, \cdots, H\}$, $k \in \mathcal{K}$ and $\gamma \in \Gamma$. More precisely, when the binary variable $\eta^k_{\gamma,i} = 1$, constraints (4.26c)-(4.26d) make sure that $0 \leq \lambda^k_{n,s} \leq 1$, and when $\eta^k_{\gamma,i} = 0$, constraints (4.26e) and $\tau^k_{n,s,\gamma,i} \geq 0$ guarantee that $\lambda^k_{n,s} = 0$. Thus, there exists an equivalence relation between the bi-linear

term $\tau^k_{n,s,\gamma,i} = \lambda^k_{n,s}\ \eta^k_{\gamma,i}$ and constraints (4.26c)-(4.26d), and $\tau^k_{n,s,\gamma,i} \geq 0$.

Next, if we do the change of variables $\tau^k_{n,s,\gamma,i} = \lambda^k_{n,s}\ \eta^k_{\gamma,i}$ in the objective function (4.38a), and replace it in the objective function (4.23), we obtain the objective function $\chi_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s}; \boldsymbol{\tau_s}, \boldsymbol{\eta_s}, \boldsymbol{\lambda_s})$ is defined for each scenario $s \in \mathcal{S}$ defined by (4.27). Therefore, using the approximation of (4.24) with the problem (4.38a)-(4.38c) and the McCormick type constraints (4.26c)-(4.26e), and $\tau^k_{n,s,\gamma,i} \geq 0$, we are able to approximate the optimal objective function value of the maximization problem $\Psi_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s})$ in (4.23) by the MILP (4.26a)-(4.26f). □

A.4. Proof of Theorem IV.2

Proof. We need to prove two things, which include that (i) the scenario cuts derived by the scenario cut-generating problem $\widetilde{\Psi}_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s})$ or (4.26a)-(4.26f) for solving the IMSDRO-APRX model are *valid*, and (ii) *finitely many* scenario cuts suffice to reach a feasible solution that satisfies constraints (4.28b).

(i) For any value of $(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s})$ satisfying constraints (4.29c) and (4.29d), the optimization problem $\Psi_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s})$ can be approximated by $\widetilde{\Psi}_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s})$ according to the result of Theorem IV.1, and so the scenario cuts (4.29b) are valid.

(ii) The scenario cut-generating problem $\widetilde{\Psi}_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s})$ is not dependent on the values of $\boldsymbol{y_s}$, $\widehat{\boldsymbol{y}}$, $\boldsymbol{\alpha_s}$, $\boldsymbol{\beta_s}$. The number of binary variables of the problem $\widetilde{\Psi}_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s})$ is limited to $|H| \times |\Gamma| \times |\mathcal{K}|$, and for any value for binary variables that satisfy $\sum_{i=0}^{H} \eta^k_{\gamma,i} = 1$, the feasible region of this optimization problem is a polyhedron with finite extreme points. Therefore, the maximum number of scenario cuts corresponding to the extreme points of the feasible region of $\widetilde{\Psi}_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s})$ for any value of binary variables are limited.

Therefore, the constraint generation Algorithm 4 terminates after finite number of iterations. □

4.9.2 Appendix B: Scenario Tree and Ambiguity Set Construction Approach.

One often starts from a full description of stochastic random variable (the number of patient referrals in the CAS problem). Solving an stochastic model with such a full description of a stochastic random variable is however next to impossible. We therefore need a *scenario construction algorithm* to translate the full representation of the stochastic variable into a set of discrete realizations (i.e., scenarios) of that stochastic variable. To adequately represent the appointment request stochastic process, we need to generate a sufficient number of scenarios; that is, the set of scenarios needs to cover the most plausible realizations of the stochastic process. This often requires a very large number of scenarios, typically generated by a scenario construction algorithm. However, the computational burden of solving such a

stochastic model with a large number of scenarios is extremely high and for practical purposes often impossible. To avoid such intractability, a *scenario reduction algorithm* is then deployed so as to reduce the cardinality of the set of scenarios. In general, the goal of a scenario construction algorithm is to minimize the error caused by the approximation of the true stochastic process with a scenario tree.

In this Appendix, we first explain our approach along with an example for how we generate a scenario tree for the number of patient arrivals in B.1, B.2 and B.3. In B.4, we pinpoint how a scenario generation and reduction can be evaluated. Finally, in B.5, we explain our method on how we generate an ambiguity set for the surgery duration.

B.1. Scenario Reduction Heuristics

We consider a T-dimensional stochastic process $\boldsymbol{\xi} = \{\xi_t\}_{t=1}^{T}$ with a distribution function F and a finite support for the number of patient appointment requests over T days. This finite support is presented by S discrete scenarios through $supp(\boldsymbol{\xi}) = \{\xi^{(1)}, \xi^{(2)}, \cdots, \xi^{(S)}\}$ where $\xi^{(s)} = \{\xi_t^{(s)}\}_{t=1}^{T}$ for $s \in \mathcal{S} = \{1, \cdots, S\}$. The corresponding scenario probability is denoted by π_s and $\sum_{s=1}^{S} \pi_s = 1$. Assume P is the distribution function of another T-dimensional stochastic process $\tilde{\boldsymbol{\xi}} = \{\tilde{\xi}_t\}_{t=1}^{T}$. Let $supp(\tilde{\boldsymbol{\xi}}) = \{\tilde{\xi}^{(1)}, \tilde{\xi}^{(2)}, \cdots, \tilde{\xi}^{(S')}\}$ where $\tilde{\xi}^{(s')} = \{\tilde{\xi}_t^{(s')}\}_{t=1}^{T}$, and S' be the number of discrete scenarios with corresponding scenario probabilities $\tilde{\pi}_{s'}$ for $s' \in \mathcal{S}' = \{1, \cdots, S'\}$ and $\sum_{s'=1}^{S'} \tilde{\pi}_{s'} = 1$.

The *Kantorovich distance* $D_T(F, P)$ between the above-mentioned stochastic processes F and P is the optimal solution of the following linear transportation problem:

$$D_T(F,P) = \min_{\rho} \Bigg\{ \sum_{s=1}^{S} \sum_{s'=1}^{S'} \rho_{s,s'} \cdot d_{|\mathcal{T}|}(\xi^{(s)}, \tilde{\xi}^{(s')}), \qquad (4.39)$$
$$s.t. \ \sum_{s=1}^{S} \rho_{s,s'} = \pi_{s'}, \ \sum_{s'=1}^{S'} \rho_{s,s'} = \tilde{\pi}_s, \ \rho_{s,s'} \geq 0, \ \forall s \in \mathcal{S}, \forall s' \in \mathcal{S}' \Bigg\},$$

where $d_t(\xi^{(s)}, \tilde{\xi}^{(s')}) = \sum_{\upsilon=1}^{t} || \ \xi_\upsilon^{(s)} - \tilde{\xi}_\upsilon^{(s')} ||$, $t \in \mathcal{T} = \{1, \cdots, T\}$, and $||.||$ is a norm function over $\mathbb{R}^T$. Thus, $d_{|\mathcal{T}|}(\xi^{(s)}, \tilde{\xi}^{(s')})$ is total distance between scenarios s and s'.

If we assume that P is a *reduced* distribution function of F, its discrete support then includes scenarios $\tilde{\xi}^{(s')} = \{\tilde{\xi}_t^{(s')}\}_{t=1}^{T}$, $s' \in \{1, \cdots, S\} \backslash del(\mathcal{S})$ where $del(\mathcal{S})$ is the set of deleted scenarios from the original scenario set $\mathcal{S}$. For a pre-specified set $del(\mathcal{S}) \subset \mathcal{S}$, the Kantorovich distance between stochastic processes F and P can be calculated by the following expression:

$$D_T(F,P) = \sum_{s \in \ del(\mathcal{S})} \pi_s \cdot \min_{s' \notin \ del(\mathcal{S})} \Big\{ d_{|\mathcal{T}|}(\xi^{(s)}, \tilde{\xi}^{(s')}) \Big\}. \qquad (4.40)$$

Moreover, the scenario probabilities $\tilde{\pi}_{s'}$, $s' \notin del(\mathcal{S})$ for the reduced set of scenarios $\{\tilde{\xi}^{(s')}\}_{s' \notin del(\mathcal{S})}$ are given by $\tilde{\pi}_{s'} = \pi_{s'} + \sum_{s \in del_{s'}(\mathcal{S})} \pi_s$, where $del_{s'}(\mathcal{S}) = \{s \in del(\mathcal{S}) : s' = s'(s)\}$, and also $s'(s) \in \arg\min_{s' \notin del(\mathcal{S})} d_{|\mathcal{T}|}(\xi^{(s)}, \tilde{\xi}^{(s')})$ for each scenario $s \in del(\mathcal{S})$ is a selection from the the index set of nearest scenarios to the scenario $\xi^{(s)}$ for all $s \in del(\mathcal{S})$. The optimal set $del(\mathcal{S})$ of deleted scenarios with cardinality $\kappa = |del(\mathcal{S})|$ is obtained by solving the following *scenario reduction problem*:

$$\min \Big\{ \sum_{s \in del(\mathcal{S})} \pi_s \cdot \min_{s' \notin del(\mathcal{S})} d_{|\mathcal{T}|}(\xi^{(s)}, \tilde{\xi}^{(s')}) \;\; s.t.\; del(\mathcal{S}) \subset \mathcal{S} = \{1, \cdots, S\}, \kappa = S - L \Big\}, \quad (4.41)$$

where $L = S - \kappa$ is the number of remaining scenarios after reduction.

[62] proved the NP-hardness of the scenario reduction problem (4.41) by showing its equivalence to the set covering problem. However, this problem can be solved efficiently for two special cases of $\kappa = 1$ (i.e., deleting one scenario), and $\kappa = S - 1$ (i.e., keeping one scenario). They proposed two heuristics called backward scenario reduction and forward scenario selection algorithms for solving the reduction problem efficiently. In the *backward reduction*, optimal deletion of one scenario is recursively repeated until deleting $\kappa = S - L$ scenarios while in the *forward selection*, optimal selection of one scenario is recursively done until achieving L scenarios.

B.2. Scenario Tree Construction Approach

In §4.6.1, Latin Hypercube Sampling (LHS) method [85] is used to construct a set of discrete scenarios for the corresponding multivariate stochastic parameters (number of patient requests) as a scenario fan. We then convert this scenario fan into a scenario tree, and use forward selection method to obtain an appropriate number of scenarios [62].

Let F be the probability distribution for a scenario fan of multivariate stochastic parameters, then each scenario $s \in \mathcal{S} = \{1, \cdots, S\}$ is presented by $\xi^{(s)} = \{\xi_0^{(s)}, \xi_1^{(s)}, \cdots, \xi_T^{(s)}\}$ with probability π_s. Since all scenarios are the same at the first node, i.e., $\xi_0^{(1)} = \xi_0^{(2)} = \cdots = \xi_0^{(S)}$ in the scenario fan, the total number of nodes is $S \times T + 1$, where $T = |\mathcal{T}|$, in the scenario fan. The goal of scenario tree construction is to generate a scenario tree with probability distribution $F_{\boldsymbol{\zeta}}$ based on the scenario fan in which the number of scenarios is reduced, and also the Kantorovich distance between F and $F_{\boldsymbol{\zeta}}$ is less than a pre-specified value $\boldsymbol{\zeta}$ (i.e., $D_T(F, F_{\boldsymbol{\zeta}}) \leq \boldsymbol{\zeta}$).

To this aim, the forward scenario reduction is used at each period $t \in \{1, \cdots, T\}$, and successive clustering of scenarios is then exploited to convert a scenario fan into a scenario tree. To construct a scenario tree with $D_T(F, F_{\boldsymbol{\zeta}}) \leq \boldsymbol{\zeta}$, at each period t, $\boldsymbol{\zeta}_t$ is considered for implementing forward scenario reduction under the criterion $\sum_{t=1}^{T} \boldsymbol{\zeta}_t \leq$

$\boldsymbol{\zeta}$. This means that at each period t, maximal reduction strategy is applied such that $\sum_{s\in\, del(\mathcal{S})} \pi_s \cdot \min_{s'\notin\, del(\mathcal{S})} d_t(\xi^{(s)}, \tilde{\xi}^{(s')}) \leq \boldsymbol{\zeta}_t$, where the distance between two scenarios $\xi^{(s)}$ and $\tilde{\xi}^{(s')}$ is calculated by $d_t(\xi^{(s)}, \tilde{\xi}^{(s')}) = \sum_{\upsilon=1}^{t} || \xi_\upsilon^{(s)} - \tilde{\xi}_\upsilon^{(s')}||$ at each period t, and $\tilde{\boldsymbol{\xi}}$ is the *reduced* version of $\boldsymbol{\xi}$ after implementing the reduction strategy.

Furthermore, we use $\boldsymbol{\zeta} = \boldsymbol{\zeta}_{rel} \cdot \boldsymbol{\zeta}_{max}$ where $0 < \boldsymbol{\zeta}_{rel} < 1$ is a constant parameter, which presents a scale for the amount of reduction in the scenario fan, and $\boldsymbol{\zeta}_{max}$ is the optimal distance between probability distribution of scenario fan and one of its scenarios with probability one. To generate the scenario tree, at each period t, $\boldsymbol{\zeta}_t$ is then computed by the following relation:

$$\boldsymbol{\zeta}_t = \frac{\boldsymbol{\zeta}}{T+1}\Big(\frac{1}{2} + \ \delta\ \big(1 - \frac{t}{T+1}\big)\Big), \quad \forall t, \tag{4.42}$$

where $\delta \in [0, 1]$ is a constant parameter, which is set to one in our implementation.

B.3. Illustration of Generating a Scenario Tree for the Number of Patient Referrals

In this section, we explain the details along with an example on how we generate a scenario tree for the number of appointment referrals that will be used in the IMSDRO-APRX model as well as the MS-MIP model. For each patient class, we first fit a Poisson probability distribution over the number of appointment referrals from that patient class. The LHS method is then used to generate a set of discrete scenarios for the corresponding multivariate stochastic parameters as a scenario fan. It is essential to efficiently reduce the number of scenarios in order to avoid computationally intractable stochastic programs. We then deploy a forward scenario tree construction heuristic to convert the scenario fan into a scenario tree, and so reduce the number of generated scenarios (see §B.2 for details). The strategy is to modify the fan of scenarios via bundling scenarios, which produces scenario trees with fewer scenarios than initial scenario fans.

The process of scenario tree construction for the number of appointment referrals in the CAS problem is illustrated by Figure 4.12 step by step. The number of final scenarios depends on a constant parameter $\boldsymbol{\zeta}_{rel}$ between zero and one, that represents a scale for the reduction amount compared with the scenario fan. To construct a scenario tree for our case study, an initial scenario fan with 100 scenarios (the most left tree in Figure 4.12) is generated for the stochastic parameters over an arrival horizon of $T = |\mathcal{T}| = 5$ periods (note that in our case study we have five business days as the arrival horizon $\mathcal{T}$), and the scenario tree construction approach is then implemented with $\boldsymbol{\zeta}_{rel} = 0.7$. Finally, a scenario tree with 14 scenarios (the most right tree in Figure 4.12) is obtained. It should be mentioned that by increasing the reduction scale $\boldsymbol{\zeta}_{rel}$, the number of obtained scenarios decreases, so the information loss increases. However, as the number of scenario decrease, we have a better

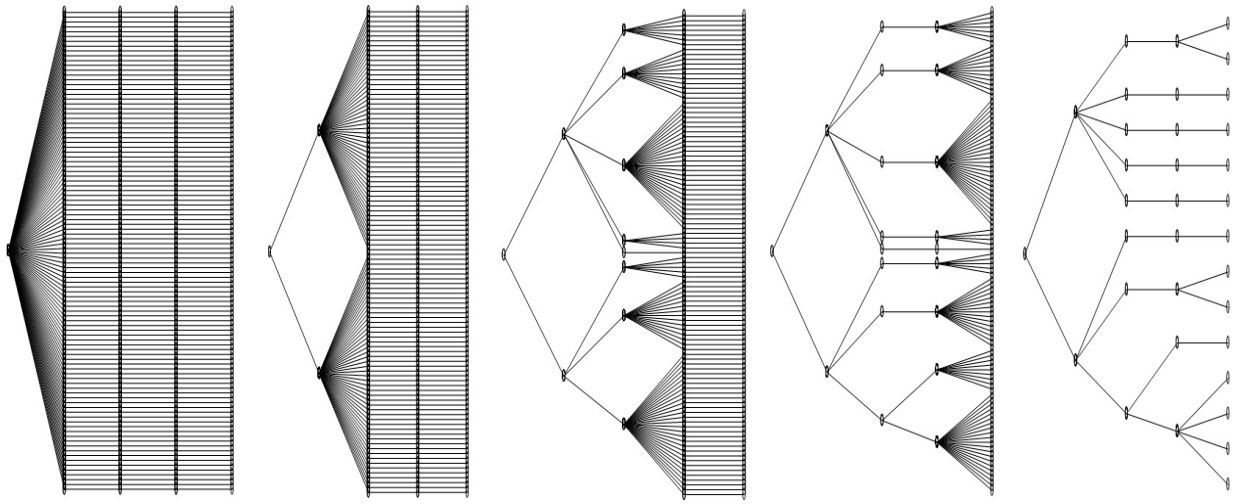


Figure 4.12: Illustration of scenario tree construction procedure for the number of appointment referrals in our case study over an arrival horizon of $T = |\mathcal{T}| = 5$ business days. We start with a scenario fan of 100 scenarios (the most left tree), and then turn it into a scenario tree of 14 scenarios (the most right tree).

computational tractability for solving the multi-stage stochastic program. Therefore, there is a trade-off between the number of scenarios and computational tractability. In §6.4, we evaluate the efficiency of this scenario construction algorithm in generating a scenario tree for the number of appointment referrals for our case study and two other random instances of our CAS problem. In particular, we find that $\boldsymbol{\zeta}_{rel} = 0.7$ is a good value for the reduction scale. Refer to §6.4 for details.

B.4. Evaluation of a Scenario Construction Algorithm.

There are two main criteria in the literature of stochastic programming [96] by which the efficiency of a scenario construction algorithm can be evaluated to ensure that there is not too much loss of information while constructing an *adequate* scenario tree. They include (i) *in-sample stability* and (ii) *out-of-sample stability*. Due to the random nature of most scenario construction algorithms (such as the LHS that we used), different scenario trees will be obtained with the same input if we apply a scenario construction algorithm multiple times. Then, the *in-sample stability* guarantees that if several scenario trees are constructed with the same input, the optimal objective function values of their corresponding stochastic optimization models with these scenario trees are the same approximately. In other words, if the objective function value does not change too much, we can claim the in-sample stability. We have done this analysis in the subsection "In-sample Stability Analysis" (see Table 4.5) in §4.6.4. Further, the *out-of-sample stability* guarantees that the objective function value obtained from implementing the scheduling policy by using our data-driven rolling-horizon

procedure should be close to the optimal objective function of the stochastic model. Indeed, the out-of-sample stability ensures that the true objective value obtained from any simulation procedure (e.g., the data-driven rolling horizon algorithm in our paper) is close to the optimal objective value of the stochastic program. This analysis is performed in the subsection "Evaluation of Objective Function Values (Overtime)" in §4.6.2 (see Table 4.3).

B.5. Ambiguity Set Generation for Surgery Durations.

We follow the procedures in the appointment scheduling literature ([55], and [91]) to generate ambiguity sets for the generation of surgery durations in the sample paths for the RHP (Algorithm 5), so that we can simulate the reality. In order to consider the distributional robustness for the surgery duration, we assume that the surgery duration can follow three classes of probability distributions: normal, gamma, and log-normal, each of which can be specified by their means and standard deviations. In each CAS problem instance for the IMSDRO-APRX model, we sample realizations $(d^1_{\gamma,k}, d^2_{\gamma,k}, \cdots, d^M_{\gamma,k})$ for each class $\gamma \in \Gamma$ and surgeon $k \in \mathcal{K}$ pair. In each of M realizations for patient class γ and surgeon $k \in \mathcal{K}$ pair, we first select randomly a distribution among normal, gamma, and log-normal, and then obtain a random surgery duration from that distribution with the known mean and standard deviation.

4.9.3 Appendix C: Alternative Optimization Model to Balance Overtime and Access Delay

In both the problem statement in §4.2 and the IMSDRO formulation in §4.3, we have assumed that (i) all patients must obtain one clinic appointment date and (if needed) one surgery appointment date within their wait time target windows, and (ii) overtime are deployed as needed to accommodate the clinical and surgical capacities. In this appendix, we relax these assumptions by trying to balance the trade-off between patients' access delays and surgeon overtimes.

To this aim, we propose an alternative optimization model for the CAS problem in which we assume no clinical and surgical overtimes for the surgeons and they only have regular clinical and surgical capacities; however, there is a penalty for the case when we cannot meet the clinical and surgical wait time targets for patients. In particular, we incur a clinic penalty of $u^{k,m}_{p,\gamma,t,s} \geq 0$ for each class $\gamma \in \Gamma$ patient $p \in \mathcal{D}^s_{\gamma,t}$ whose request is received on day $t \in \mathcal{T}$ under scenario $s \in \mathcal{S}$ and has a clinic appointment visit on $m > t + WTS_\gamma - CSG_\gamma$ (recall that the safe range for clinic appointment visit is $m \in \big[t + WTC_\gamma, t + WTS_\gamma - CSG\gamma\big]$). Moreover, we incur a surgery penalty of $v^{k,n}_{\gamma,t,m,s} \geq 0$ for class $\gamma \in \Gamma$ patients whose requests are received on day $t \in \mathcal{U} \cup \mathcal{T}$ under $s \in \mathcal{S}$, and have clinic visit on $m \in \mathcal{T} \backslash \{t_0\}$, but surgery

visit on $n > t + WTS_\gamma$ with surgeon $k \in \mathcal{K}$. We have a similar surgery penalty $e^{k,n}_{\gamma,t,t_0} \geq 0$ for class $\gamma \in \Gamma$ patients whose requests are received on day $t \in \mathcal{U} \cup \{t_0\}$, and have clinic visit on current day t_0, but surgery visit on $n > t + WTS_\gamma$ with surgeon $k \in \mathcal{K}$ (recall that the safe range for surgery appointment visit is $n \in \big[t + WTC_\gamma, t + WTS_\gamma - CSG\gamma\big]$). We summarize the new notations in Table 10. The other notations are as before (see Table 4.1).

Multi-stage stochastic model. With these three new penalty decisions variables, the

MS-MIP model (4.1)-(4.17) is turned into the following optimization model:

$$\min \sum_{s\in\mathcal{S}} \pi_s \sum_{k\in\mathcal{K}} \sum_{\gamma\in\Gamma} \Big(\sum_{m\in\mathcal{L}} \sum_{t\in\mathcal{T}} \sum_{p\in\mathcal{D}^s_{\gamma,t}} u^{k,m}_{p,\gamma,t,s} + \sum_{m\in\mathcal{L}} \sum_{t\in\mathcal{U}\cup\mathcal{T}} \sum_{n\in\mathcal{L}} v^{k,n}_{\gamma,t,m,s} + \sum_{n\in\mathcal{L}} \sum_{t\in\mathcal{U}\cup\{t_0\}} e^{k,n}_{\gamma,t,t_0} \Big) \tag{4.43a}$$

$$\text{s.t. } x^{k,m}_{p,\gamma,t,s}(m-t-WTS_\gamma+CSG_\gamma) \le u^{k,m}_{p,\gamma,t,s},\ \forall \gamma\in\Gamma, t\in\mathcal{T}, s\in\mathcal{S}, p\in\mathcal{D}^s_{\gamma,t}, k\in\mathcal{K}, m\in\mathcal{L}, \tag{4.43b}$$

$$y^{k,n}_{\gamma,t,m,s}(n-t-WTS_\gamma) \le v^{k,n}_{\gamma,t,m,s},\ \forall \gamma\in\Gamma, t\in\mathcal{T}\cup\mathcal{U}, s\in\mathcal{S}, k\in\mathcal{K}, m\in\mathcal{T}\backslash\{t_0\}, \tag{4.43c}$$

$$\widehat{y}^{k,n}_{\gamma,t,t_0}(n-t-WTS_\gamma) \le e^{k,n}_{\gamma,t,t_0},\ \forall \gamma\in\Gamma, t\in\mathcal{U}\cup\{t_0\}, k\in\mathcal{K}, n\in\mathcal{L}, \tag{4.43d}$$

$$x^{k,m}_{p,\gamma,t,s} = 0,\ \forall \gamma\in\Gamma, t\in\mathcal{T}, s\in\mathcal{S}, p\in\mathcal{D}^s_{\gamma,t}, k\in\mathcal{K}, m\in\big[t_0, t+WTC_\gamma-1\big], \tag{4.43e}$$

$$\sum_{m=t+WTC_\gamma}^{t_e} \sum_{k\in\mathcal{K}} x^{k,m}_{p,\gamma,t,s} = 1,\ \forall \gamma\in\Gamma, t\in\mathcal{T}, s\in\mathcal{S}, p\in\mathcal{D}^s_{\gamma,t}, \tag{4.43f}$$

$$r_\gamma\Big(\sum_{p\in\widetilde{\mathcal{D}}_{\gamma,t}} \widetilde{x}^{k,m}_{p,\gamma,t} \Big) \le \sum_{n=m+CSG_\gamma}^{t_e} y^{k,n}_{\gamma,t,m,s},\ \forall \gamma\in\Gamma, t\in\mathcal{U}, m\in\mathcal{T}\backslash\{t_0\}, k\in\mathcal{K}, s\in\mathcal{S}, \tag{4.43g}$$

$$r_\gamma\Big(\sum_{p\in\mathcal{D}^s_{\gamma,t}} x^{k,m}_{p,\gamma,t,s} \Big) \le \sum_{n=m+CSG_\gamma}^{t_e} y^{k,n}_{\gamma,t,m,s},\ \forall \gamma\in\Gamma, t\in\mathcal{T}, m\in\mathcal{T}\backslash\{t_0\}, k\in\mathcal{K}, s\in\mathcal{S}, \tag{4.43h}$$

$$\widetilde{z}^{k}_{\gamma,t} \le \sum_{n=t_0+CSG_\gamma}^{t_e} \widehat{y}^{k,n}_{\gamma,t,t_0},\ \forall \gamma\in\Gamma, t\in\mathcal{U}\cup\{t_0\}, k\in\mathcal{K}, \tag{4.43i}$$

$$\sum_{\gamma\in\Gamma} c_\gamma \Big(\sum_{t\in\mathcal{U}} \sum_{p\in\widetilde{\mathcal{D}}_{\gamma,t}} \widetilde{x}^{k,m}_{p,\gamma,t} + \sum_{t\in\mathcal{T}} \sum_{p\in\mathcal{D}^s_{\gamma,t}} x^{k,m}_{p,\gamma,t,s} \Big) \le U^k_m,\ \forall m\in\mathcal{L}, k\in\mathcal{K}, s\in\mathcal{S}, \tag{4.43j}$$

$$\sum_{\gamma\in\Gamma} d_{\gamma,k} \Big(\sum_{t\in\mathcal{U}} \sum_{m\in\mathcal{U}} \widetilde{y}^{k,n}_{\gamma,t,m} + \sum_{t\in\mathcal{T}} \sum_{m\in\mathcal{T}\backslash\{t_0\}} y^{k,n}_{\gamma,t,m,s} + \sum_{t\in\mathcal{T}\cup\{t_0\}} \widehat{y}^{k,n}_{\gamma,t,t_0} \Big) \le V^k_n, \forall n\in\mathcal{L}, k\in\mathcal{K}, s\in\mathcal{S}, \tag{4.43k}$$

$$u^{k,m}_{p,\gamma,t,s} \ge 0,\ \forall \gamma\in\Gamma, t\in\mathcal{T}, s\in\mathcal{S}, p\in\mathcal{D}^s_{\gamma,t}, k\in\mathcal{K}, m\in\mathcal{L}, \tag{4.43l}$$

$$v^{k,n}_{\gamma,t,m,s} \ge 0,\ \forall \gamma\in\Gamma, t\in\mathcal{T}\cup\mathcal{U}, s\in\mathcal{S}, k\in\mathcal{K}, m\in\mathcal{T}\backslash\{t_0\}, \tag{4.43m}$$

$$e^{k,n}_{\gamma,t,t_0} \ge 0,\ \forall \gamma\in\Gamma, t\in\mathcal{U}\cup\{t_0\}, k\in\mathcal{K}, n\in\mathcal{L}, \tag{4.43n}$$

$$(4.12)-(4.13), (4.14)-(4.17). \tag{4.43o}$$

The objective function (4.43a) is to minimize the expected penalties due to not meeting clinical and surgical wait time targets for patients. Constraints (4.43b)-(4.43d) along with constraints (4.43l)-(4.43n) are the related constraints for making the penalty decisions $u^{k,m}_{p,\gamma,t,s}$, $v^{k,n}_{\gamma,t,m,s}$ and $e^{k,n}_{\gamma,t,t_0}$. Constraints (4.43e)-(4.43f) and (4.43g)-(4.43i) determine the clinic and

surgery appointment visits, respectively. Constraints (4.43j)-(4.43k) restricts the regular clinical and surgical capacities of surgeons on each day, respectively. Similar to the MS-MIP model (4.1)-(4.17), we assume that the surgery durations are deterministic in the above MS-MIP model (4.43a)-(4.43o).

Stage Decision Variables	
$u^{k,m}_{p,\gamma,t,s}$	: Clinical penalty if a class $\gamma \in \Gamma$ patient p whose request is received on day $t \in \mathcal{T}$ under scenario $s \in \mathcal{S}$, has clinic visit on day $m > t + WTS_\gamma - CSG_\gamma$ with surgeon $k \in \mathcal{K}$.
$v^{k,n}_{\gamma,t,m,s}$	: Surgical penalty if class $\gamma \in \Gamma$ patients whose requests are received on day $t \in \mathcal{U} \cup \mathcal{T}$ under $s \in \mathcal{S}$ and have clinic visit on $m \in \mathcal{T} \backslash \{t_0\}$, have surgery visit on day $n > t + WTS_\gamma$ with surgeon $k \in \mathcal{K}$.
$e^{k,n}_{\gamma,t,t_0}$	: Surgical penalty if class $\gamma \in \Gamma$ patients whose requests are received on day $t \in \mathcal{U} \cup \{t_0\}$, and have clinic visit on day t_0, have surgery visit on $n > t + WTS_\gamma$ with surgeon $k \in \mathcal{K}$.

Table 4.8: The description of new notations used by the MS-MIP model (4.43a)-(4.43o) of the CAS problem.

Integrated multi-stage stochastic and distributionally robust model. To model the uncertainty in surgery durations presented by the moment-based ambiguity set $\Phi(\boldsymbol{\mu}, \boldsymbol{\sigma}, \Theta)$ in (4.19a)-(4.19c), we deploy the IMSDRO approach described in §4.3.2, to MS-MIP model (4.43a)-(4.43o). The resulting IMSDRO model is formulated as follows:

$$\bar{Z}^{IMSDRO} = \min_{\boldsymbol{x},\boldsymbol{y},\widehat{\boldsymbol{y}},\boldsymbol{u},\boldsymbol{v},\boldsymbol{e}} \Big\{ w_1 \Big(\sum_{s\in\mathcal{S}} \pi_s \sum_{k\in\mathcal{K}} \sum_{\gamma\in\Gamma} \Big(\sum_{m\in\mathcal{L}} \sum_{t\in\mathcal{T}} \sum_{p\in\mathcal{D}^s_{\gamma,t}} u^{k,m}_{p,\gamma,t,s} + \sum_{m\in\mathcal{L}} \sum_{t\in\mathcal{U}\cup\mathcal{T}} \sum_{n\in\mathcal{L}} v^{k,n}_{\gamma,t,m,s} + \sum_{n\in\mathcal{L}} \sum_{t\in\mathcal{U}\cup\{t_0\}} e^{k,n}_{\gamma,t,t_0} \Big) \Big) + w_2 \Big(\sum_{s\in\mathcal{S}} \pi_s \max_{P\in\Phi(\boldsymbol{\mu},\boldsymbol{\sigma},\Theta)} \mathbb{E}_P \Big[f_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{d}) \Big] \Big) \Big\}, \tag{4.44a}$$

$$\text{s.t. } (\boldsymbol{x_s}, \boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{u_s}, \boldsymbol{v_s}, \boldsymbol{e}) \in \mathcal{O}_s, \quad \forall s \in \mathcal{S} \tag{4.44b}$$

where $\mathcal{O}_s$ is the feasible region defined by the constraints (4.43a)-(4.43j) and (4.43l)-(4.43o). The objective function (4.44a) is obtained by removing constraints (4.43k) from the MS-MIP model (4.43a)-(4.43o) and adding its worst-case expected value over the set of plausible surgery duration distributions $P \in \Phi(\boldsymbol{\mu}, \boldsymbol{\sigma}, \Theta)$ into the objective function, which results in the min-max IMSDRO model (4.44a)-(4.44b).

An important feature of IMSDRO model (4.44a)-(4.44b) compared with IMSDRO model (4.20a)-(4.20b) is that two parts of the objective functions (4.44a) weighted by w_1 and w_2 are in fact two *conflicting objectives*. The first part weighted by w_1 is to minimize the expected penalties incurred due to not meeting the clinical and surgical wait time targets for patients, and the second one weighted by w_2 is to minimize the maximum penalties incurred due to not satisfying the regular surgical capacities of surgeons. Indeed, it is the case that either

we can meet the clinical and surgical wait time targets for patients, or we can make regular capacities for surgeons.

Following the steps of IMSDRO approach described in Propositions 11 and 12 and Theorem IV.1, the min-max IMSDRO model (4.44a)-(4.44b) can be approximated by the following optimization model:

$$\tilde{\bar{Z}}^{IMSDRO} = \min_{\boldsymbol{x},\boldsymbol{y},\widehat{\boldsymbol{y}},\boldsymbol{u},\boldsymbol{v},\boldsymbol{e},\boldsymbol{\delta},\boldsymbol{\alpha},\boldsymbol{\beta}} \Big\{ w_1 \Big(\sum_{s\in\mathcal{S}} \pi_s \sum_{k\in\mathcal{K}} \sum_{\gamma\in\Gamma} \Big(\sum_{m\in\mathcal{L}} \sum_{t\in\mathcal{T}} \sum_{p\in\mathcal{D}^s_{\gamma,t}} u^{k,m}_{p,\gamma,t,s} + \sum_{m\in\mathcal{L}} \sum_{t\in\mathcal{U}\cup\mathcal{T}} \sum_{n\in\mathcal{L}} v^{k,n}_{\gamma,t,m,s} + \sum_{n\in\mathcal{L}} \sum_{t\in\mathcal{U}\cup\{t_0\}} e^{k,n}_{\gamma,t,t_0} \Big) \Big) + w_2 \Big(\sum_{s\in\mathcal{S}} \pi_s \sum_{\gamma\in\Gamma} \sum_{k\in\mathcal{K}} \big(\mu_{\gamma,k}\, \alpha^k_{\gamma,s} + (\mu^2_{\gamma,k} + \sigma^2_{\gamma,k})\, \beta^k_{\gamma,s} \big) + \sum_{s\in\mathcal{S}} \pi_s \delta_s \Big) \Big\}, \tag{4.45a}$$

$$\text{s.t. } \delta_s \geq \widetilde{\Psi}_s(\boldsymbol{y_s}, \widehat{\boldsymbol{y}}, \boldsymbol{\alpha_s}, \boldsymbol{\beta_s}), \quad \forall s \in \mathcal{S} \tag{4.45b}$$

$$(\boldsymbol{x_s},\ \boldsymbol{y_s}, \widehat{\boldsymbol{y}},\ \boldsymbol{u_s},\ \boldsymbol{v_s},\ \boldsymbol{e}) \in \mathcal{O}_s, \quad \forall s \in \mathcal{S} \tag{4.45c}$$

$$\delta_s \in \mathbb{R},\ \boldsymbol{\alpha_s},\ \boldsymbol{\beta_s} \in \mathbb{R}^{|\Gamma|\times|\mathcal{K}|}, \quad \forall s \in \mathcal{S}. \tag{4.45d}$$

The above IMSDRO-APRX model (4.45a)-(4.45d) can be solved by the constraint generation Algorithm 4 described in §4.4.

CHAPTER V

Conclusions and Future Research

5.1 Summary and Conclusions

This dissertation developed different personalized data-driven learning and optimization methods for different applications, including chronic disease management, online appointment scheduling platforms, and healthcare delivery systems.

Chapter II introduced a new personalized disease progression control model with a two-dimensional nested control, and proposed the first contextual learning and optimization algorithm for it. For this algorithm, we provided a rigorous analytical performance analysis that involves several new technical ideas integrating the strength of contextual bandit and online convex optimization in a seamless fashion. Although our model and algorithm were motivated by a fundamental medical decision-making problem, they can also be applied to a wide range of other operation problems (e.g., joint inventory and pricing/vehicle routing) in which there are two levels of decision-making process. We illustrated our algorithm's practical relevance by evaluating it empirically on a critical chronic disease problem of controlling high blood pressure for patients with Type 2 Diabetes Mellitus (T2DM) at high risk for CVD. Our algorithm/model fills an important gap in current clinical guidelines of blood pressure management, namely that they do not inform the choice and dosage for the third-line BP-lowering medication to maintain the target SBP of 120 mmHg. Our empirical results provide medical professionals with critical insights into the effect of different third-line medications on achieving the BP target of 120 mmHg for patients with T2DM at high risk for CVD. Instead of a trial-and-error approach to find the right medication and its corresponding dosage to achieve BP control, which is common in everyday clinical practice, our decision support tool provides an optimized medication and dosage considering all key contextual variables of a given individual. Personalizing BP treatment, in particular for patients with diabetes who are at increased risk of cardiovascular and microvascular events, can improve outcomes and result in cost containment by averting some of these costly events.

Chapter III studied an important class of online scheduling problem with budgeted overtime in which the model has no knowledge about the pattern or underlying distribution of the arrival process. Upon arrival of a customer, the system makes an instantaneous and irrevocable allocation decision for this customer, without knowing any information on subsequent customers. We adopt a primal-dual approach to develop new effective and efficient online algorithms to make for every arriving patient on every day in the horizon not only a date-server allocation decision but also a decision of whether or not to use overtime to serve the patient. The proposed online policies (i) are robust to future uncertain information, (ii) are easy to implement and extremely efficient to compute, (iii) allow for heterogeneity in both reward and service requirement by a server, and (iv) admit a theoretical performance guarantee. Comparing our online policy with the optimal offline policy, we obtain a competitive ratio which guarantees the worst-case performance of our proposed online policy. For practical settings, we extend our online algorithm to a rolling horizon paradigm. A particular emphasis of this paper has been put on the real-world applicability of our proposed methods. The online resource allocation problem studied in this work is not only investigated through a theoretical lens but also from the perspective of healthcare operations. We evaluate the empirical performance of our online algorithms by using real appointment-scheduling data from a healthcare clinic of our partner health system. Our computational results show that the proposed online policies perform much better than their theoretical worst-case performance guarantee and extremely well compared to the pervasive FCFS scheduling heuristic and a new policy we term the nested threshold policy.

Chapter IV studied a new class of appointment scheduling problems called the coordinated clinic and surgery appointment scheduling" in which patients are stratified into patient classes, with limits on the allowable access delay from request to appointment dates. We introduced the concept of care coordination in the sense of setting appointments for pairs of sequential clinic and (if needed) surgery visits that together achieve timely access to care. Methodologically speaking, our integrated multi-stage stochastic and distributionally robust optimization (IMSDRO) is the first optimization approach that can jointly incorporate different types of uncertainty in the number of patient appointment requests by a scenario tree, and in surgery durations by a moment-based ambiguity set for distributional robustness. Using the special structure of the CAS problem, we proposed a constraint generation algorithm for efficiently solving this problem. We then developed a new data-driven rolling horizon procedure to implement the decisions made by the IMSDRO approach in practice. This allows healthcare practitioners to make efficient use of data that is obtained as time unfolds, and so adjust their decisions in a rolling horizon framework. In a sense, our methods/models can be applied in an online (or real-time) fashion. We tested the validity of our mod-

els/algorithms in a case study of scheduling clinic consultation and surgery appointments, and demonstrated that a significant improvement could be achieved if our partner hospital were to switch from the current heuristic scheduling protocol to our proposed policies. We provide several practical insights from our empirical analysis as well.

5.2 Future Research

In summary, we addressed three major areas surrounding data-driven learning and optimization; however, several more avenues of research, with both methodological contribution and practical impact, can be conducted to build on this thesis.

Unlike the setting of our problem in Chapter II, a patient often has to be treated sequentially over multiple stages. For such setting, dynamic treatment regimes (DTR) provide a multi-stage personalized framework of distinct decision rules that determines a treatment decision at each stage given the patient's evolving condition. While most literature focuses on learning the optimal DTR from offline historical data, a promising future direction is to develop adaptive online reinforcement learning (RL) algorithms that achieve near-optimal regret for DTRs in online settings, without access to historical data. The next future research is to develop online RL algorithms that learn the optimal DTR while leveraging any imperfect and confounded offline historical data available. They are useful in ongoing monitoring of patient's condition and just-in-time interventions.

There are some limitations in the proposed models in Chapter III that could spur future research. First, we do not consider stochasticity in the service time requirement in our theoretical analysis (even though we investigate it empirically). Thus, it would be interesting to see if one can design an online algorithm that can handle stochastic service times and obtain a CR. Second, the rolling horizon extension is myopic and not optimal. It would be interesting to study the full-blown dynamic optimization problem and establish meaningful theoretical results. Third, in practice, patient no-shows and cancellations happen from time to time. No-shows do not impact the model at the daily level (only time of day). The current methodology does not incorporate cancellations, and we leave it for future research. Lastly, there is a small gap between our lower and upper bounds for the competitive ratio of our online algorithms, so it is not tight. Thus, whether the proposed online primal-dual algorithms admit a tighter lower bound or there is a tighter upper bound for any online algorithm remains a question for future research.

This study in Chapter IV has a few limitations. In our models, we do not consider patient no-shows and cancellations as well as the potential seasonality in demand as they rarely happen in our highly-specialized partner surgical suites. Patient preferences are also

not part of our models and algorithms. Clearly, in many health care environments, the patient can prioritize the selection of the provider with whom they feel most comfortable. Our scope is; however, limited to the important class of environments in which the patients typically accept the provider offering the earliest access. Moreover, the allocation of resources including operating rooms to surgeons is not the main focus in our paper. Finally, given that a tractable system state can be defined, approximate or robust dynamic programming approaches may be used to solve the CAS problem. These ideas could be promising future research directions in the area of appointment scheduling.

BIBLIOGRAPHY

[1] Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In *Proceedings of the 24th International Conference on Neural Information Processing Systems*, NIPS'11, pages 2312–2320, Red Hook, NY, USA, 2011. Curran Associates Inc.

[2] Marc Abeille, Alessandro Lazaric, et al. Linear thompson sampling revisited. *Electronic Journal of Statistics*, 11(2):5165–5197, 2017.

[3] ADA. 10. cardiovascular disease and risk management: Standards of medical care in diabetes2019. *Diabetes Care*, 42(Supplement 1):S103–S123, 2019.

[4] Gagan Aggarwal, Gagan Goel, Chinmay Karande, and Aranyak Mehta. Online vertex-weighted bipartite matching and single-bid budgeted allocations. In *Proceedings of the twenty-second annual ACM-SIAM symposium on Discrete Algorithms*, pages 1253–1264. Society for Industrial and Applied Mathematics, 2011.

[5] Shipra Agrawal and Nikhil R. Devanur. Linear contextual bandits with knapsacks. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, page 34583467, Red Hook, NY, USA, 2016. Curran Associates Inc.

[6] Shipra Agrawal and Navin Goyal. Thompson sampling for contextual bandits with linear payoffs. In *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28 III*, ICML'13, pages 1220–1228, Atlanta, GA, USA, 2013. JMLR.org.

[7] Amir Ahmadi-Javid, Zahra Jalali, and Kenneth J Klassen. Outpatient appointment systems in healthcare: A review of optimization studies. *European Journal of Operational Research*, 258(1):3–34, 2017.

[8] Vishal Ahuja and John R Birge. Response-adaptive designs for clinical trials: Simultaneous learning from multiple patients. *European Journal of Operational Research*, 248(2):619–633, 2016.

[9] Oguzhan Alagoz, Lisa M Maillart, Andrew J Schaefer, and Mark S Roberts. The optimal timing of living-donor liver transplantation. *Management Science*, 50(10):1420–1430, 2004.

[10] Jeffrey M Alden and Robert L Smith. Rolling horizon procedures in nonhomogeneous markov decision processes. *Operations Research*, 40(3-supplement-2):S183–S194, 1992.

[11] Daniel Almirall, Scott N Compton, Meredith Gunlicks-Stoessel, Naihua Duan, and Susan A Murphy. Designing a pilot sequential multiple assignment randomized trial for developing an adaptive treatment strategy. *Statistics in medicine*, 31(17):1887–1902, 2012.

[12] Arielle Anderer, Hamsa Bastani, and John Silberholz. Adaptive clinical trial designs with surrogates: When should we bother? *Working Paper, University of Michigan, Ann Arbor, MI, Available at SSRN3397464*, 2019.

[13] Lawrence J Appel, Jackson T Wright Jr, Tom Greene, Lawrence Y Agodoa, Brad C Astor, George L Bakris, William H Cleveland, Jeanne Charleston, Gabriel Contreras, Marquetta L Faulkner, et al. Intensive blood-pressure control in hypertensive chronic kidney disease. *New England Journal of Medicine*, 363(10):918–929, 2010.

[14] Carlos Arauz-Pacheco, Marian A Parrott, and Phillip Raskin. Hypertension management in adults with diabetes. *Diabetes care*, 27:S65, 2004.

[15] Peter Auer. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3(11):397–422, 2002.

[16] Turgay Ayer, Oguzhan Alagoz, and Natasha K Stout. Or foruma pomdp approach to personalize mammography screening decisions. *Operations Research*, 60(5):1019–1034, 2012.

[17] Nur Ayvaz and Woonghee Tim Huh. Allocation of hospital capacity to multiple types of patients. *Journal of Revenue and Pricing Management*, 9(5):386–398, 2010.

[18] Moshe Babaioff, Nicole Immorlica, David Kempe, and Robert Kleinberg. Online auctions and generalized secretary problems. *ACM SIGecom Exchanges*, 7(2):7, 2008.

[19] Bahman Bahmani and Michael Kapralov. Improved bounds for online stochastic matching. *Algorithms–ESA 2010*, pages 170–181, 2010.

[20] Santiago Balseiro, Negin Golrezaei, Mohammad Mahdian, Vahab Mirrokni, and Jon Schneider. Contextual bandits with cross-learning. *Working Paper, Columbia University, New York, NY, arXiv preprint arXiv:1809.09582*, 2018.

[21] Hamsa Bastani and Mohsen Bayati. Online decision making with high-dimensional covariates. *Operations Research*, 68(1):276–294, 2020.

[22] Hamsa Bastani, Mohsen Bayati, and Khashayar Khosravi. Mostly exploration-free algorithms for contextual bandits. *Forthcoming in Management Science*, 2020.

[23] Leanne Bellamy, Juan-Pablo Casas, Aroon D Hingorani, and David Williams. Type 2 diabetes mellitus after gestational diabetes: a systematic review and meta-analysis. *The Lancet*, 373(9677):1773–1779, 2009.

[24] Emelia J Benjamin, Salim S Virani, Clifton W Callaway, Alanna M Chamberlain, Alexander R Chang, Susan Cheng, Stephanie E Chiuve, Mary Cushman, Francesca N Delling, Rajat Deo, et al. Heart disease and stroke statistics-2018 update: a report from the american heart association. *Circulation*, 137(12):e67–e492, 2018.

[25] Bjorn P Berg, Brian T Denton, S Ayca Erdogan, Thomas Rohleder, and Todd Huschka. Optimal booking and scheduling in outpatient procedure centers. *Computers & Operations Research*, 50:24–37, 2014.

[26] Donald A Berry. Adaptive clinical trials in oncology. *Nature reviews Clinical oncology*, 9(4):199, 2012.

[27] D Bertsekas. Rollout algorithms for constrained dynamic programming. *Lab. for Information and Decision Systems Report*, 2646, 2005.

[28] Dimitri P Bertsekas and David A Castanon. Rollout algorithms for stochastic scheduling problems. *Journal of Heuristics*, 5(1):89–108, 1999.

[29] Dimitri P Bertsekas, John N Tsitsiklis, and Cynara Wu. Rollout algorithms for combinatorial optimization. *Journal of Heuristics*, 3(3):245–262, 1997.

[30] Dimitris Bertsimas, Allison OHair, Stephen Relyea, and John Silberholz. An analytics approach to designing combination chemotherapy regimens for cancer. *Management Science*, 62(5):1511–1531, 2016.

[31] Dimitris Bertsimas and Ying Daisy Zhuo. Novel target discovery of existing therapies: Path to personalized cancer therapy. *Informs Journal on Optimization*, 2(1):1–13, 2020.

[32] John R Birge and Francois Louveaux. *Introduction to stochastic programming*. Springer Science & Business Media, 2011.

[33] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer-Verlag, Berlin, Heidelberg, 2006.

[34] Allan Borodin and Ran El-Yaniv. *Online computation and competitive analysis*. cambridge university press, 2005.

[35] Djallel Bouneffouf and Irina Rish. A survey on practical applications of multi-armed and contextual bandits. *Working Paper, IBM Thomas J. Watson Research Center, Yorktown Heights, NY, Available at arXiv preprint arXiv:1904.10040*, 2019.

[36] Jason Brinkley. A doubly robust estimator for the attributable benefit of a treatment regime. *Statistics in medicine*, 33(29):5057–5073, 2014.

[37] Cameron B Browne, Edward Powley, Daniel Whitehouse, Simon M Lucas, Peter I Cowling, Philipp Rohlfshagen, Stephen Tavener, Diego Perez, Spyridon Samothrakis, and Simon Colton. A survey of monte carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in games*, 4(1):1–43, 2012.

[38] Sébastien Bubeck, Rémi Munos, Gilles Stoltz, and Csaba Szepesvári. X-armed bandits. *Journal of Machine Learning Research*, 12(5):1655–1695, 2011.

[39] Niv Buchbinder, Kamal Jain, and Joseph Naor. Online primal-dual algorithms for maximizing ad-auctions revenue. *Algorithms–ESA 2007*, pages 253–264, 2007.

[40] Niv Buchbinder, Tracy Kimbrel, Retsef Levi, Konstantin Makarychev, and Maxim Sviridenko. Online make-to-order joint replenishment model: Primal-dual competitive algorithms. *Operations Research*, 61(4):1014–1029, 2013.

[41] Niv Buchbinder and Joseph Naor. The design of competitive online algorithms via a primal–dual approach. *Foundations and Trends® in Theoretical Computer Science*, 3(2–3):93–263, 2009.

[42] Tim Carnes and David B Shmoys. Primal-dual schema for capacitated covering problems. *Mathematical Programming*, 153(2):289–308, 2015.

[43] Ferran Catala-Lopez, Diego Macías Saint-Gerons, Diana Gonzalez-Bermejo, Giuseppe M Rosano, Barry R Davis, Manuel Ridao, Abel Zaragoza, Dolores Montero-Corominas, Aurelio Tobias, Cesar de la Fuente-Honrubia, et al. Cardiovascular and renal outcomes of renin–angiotensin system blockade in adult patients with diabetes mellitus: A systematic review with network meta-analyses. *PLoS medicine*, 13(3):e1001971, 2016.

[44] Tugba Cayirli and Emre Veral. Outpatient scheduling in health care: a review of literature. *Production and operations management*, 12(4):519–549, 2003.

[45] Nicolo Cesa-Bianchi, Yoav Freund, David Haussler, David P Helmbold, Robert E Schapire, and Manfred K Warmuth. How to use expert advice. *Journal of the ACM*, 44(3):427–485, 1997.

[46] Carri W Chan, Linda V Green, Yina Lu, Nicole Leahy, and Roger Yurt. Prioritizing burn-injured patients during a disaster. *Manufacturing & Service Operations Management*, 15(2):170–190, 2013.

[47] Richard E Chatwin. On the upper bound for online allocation with concave returns. Working paper, Stanford University, CA, 2017.

[48] Wang Chi Cheung, David Simchi-Levi, and Ruihao Zhu. Hedging the drift: Learning to optimize under non-stationarity. *Working Paper, MIT, Cambridge, MA, Available at arXiv preprint arXiv:1903.01461*, 2019.

[49] Stephen Chick, Martin Forster, and Paolo Pertile. A bayesian decision-theoretic model of sequential experimentation with delayed response. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, pages 1439–1462, 2017.

[50] Stephen E Chick, Noah Gans, and Ozge Yapar. Bayesian sequential learning for clinical trials of multiple correlated medical interventions. *Working Paper, INSEAD, France*, 2018.

[51] Shein-Chung Chow. Adaptive clinical trial design. *Annual review of medicine*, 65:405–415, 2014.

[52] Wei Chu, Lihong Li, Lev Reyzin, and Robert Schapire. Contextual bandits with linear payoff functions. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, AISTATS'11, pages 208–214, Ft. Lauderdale, FL, 2011. JLMR.

[53] Varsha Dani, Thomas P Hayes, and Sham M Kakade. Stochastic linear optimization under bandit feedback. *Working Paper, University of Chicago, Chicago, IL*, 2008.

[54] Jivan Deglise-Hawkinson, Jonathan E Helm, Todd Huschka, David L Kaufman, and Mark P Van Oyen. A capacity allocation planning model for integrated care and access management. *Production and operations management*, 27(12):2270–2290, 2018.

[55] Brian Denton and Diwakar Gupta. A sequential bounding approach for optimal appointment scheduling. *IIE transactions*, 35(11):1003–1016, 2003.

[56] Brian Denton, James Viapiano, and Andrea Vogl. Optimization of surgery sequencing and scheduling decisions under uncertainty. *Health care management science*, 10(1):13–24, 2007.

[57] Brian T Denton. Optimization of sequential decision making for chronic diseases: From data to decisions. In *Recent Advances in Optimization and Modeling of Contemporary Problems*, pages 316–348. INFORMS, 2018.

[58] Nikhil R Devanur and Kamal Jain. Online matching with concave returns. In *Proceedings of the Forty-fourth Annual ACM Symposium on Theory of Computing*, pages 137–144. ACM, 2012.

[59] Nikhil R Devanur, Kamal Jain, Balasubramanian Sivan, and Christopher A Wilkens. Near optimal online algorithms and fast approximation algorithms for resource allocation problems. In *Proceedings of the 12th ACM conference on Electronic commerce*, pages 29–38. ACM, 2011.

[60] Adam Diamant, Joseph Milner, and Fayez Quereshy. Dynamic patient scheduling for multi-appointment health care programs. *Production and Operations Management*, 27(1):58–79, 2018.

[61] Jitka Dupačová. Multistage stochastic programs: The state-of-the-art and selected bibliography. *Kybernetika*, 31(2):151–174, 1995.

[62] Jitka Dupačová, Nicole Gröwe-Kuska, and Werner Römisch. Scenario reduction in stochastic programming. *Mathematical programming*, 95(3):493–511, 2003.

[63] S Ayca Erdogan and Brian Denton. Dynamic appointment scheduling of a stochastic server with uncertain demand. *INFORMS Journal on Computing*, 25(1):116–132, 2013.

[64] Jacob Feldman, Nan Liu, Huseyin Topaloglu, and Serhan Ziya. Appointment scheduling under patient preference and no-show behavior. *Operations Research*, 62(4):794–811, 2014.

[65] Jon Feldman, Monika Henzinger, Nitish Korula, Vahab S Mirrokni, and Cliff Stein. Online stochastic packing applied to display ad allocation. In *European Symposium on Algorithms*, pages 182–194. Springer, 2010.

[66] Jon Feldman, Aranyak Mehta, Vahab Mirrokni, and S Muthukrishnan. Online stochastic matching: Beating 1-1/e. In *Foundations of Computer Science, 2009. FOCS'09. 50th Annual IEEE Symposium on*, pages 117–126. IEEE, 2009.

[67] Kris Johnson Ferreira, David Simchi-Levi, and He Wang. Online network revenue management using thompson sampling. *Operations research*, 66(6):1586–1602, 2018.

[68] Sarah Filippi, Olivier Cappé, Aurélien Garivier, and Csaba Szepesvári. Parametric bandits: The generalized linear case. In *Proceedings of the 23rd International Conference on Neural Information Processing Systems - Volume 1*, NIPS'10, pages 586–594, Red Hook, NY, 2010. Curran Associates Inc.

[69] Abraham D. Flaxman, Adam Tauman Kalai, and H. Brendan McMahan. Online convex optimization in the bandit setting: Gradient descent without a gradient. In *Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '05, pages 385–394, USA, 2005. Society for Industrial and Applied Mathematics.

[70] Guillermo Gallego, Anran Li, Van-Anh Truong, and Xinshang Wang. Approximation algorithms for product framing and pricing. Working paper, Columbia University, NY, 2019.

[71] Sylvain Gelly, Levente Kocsis, Marc Schoenauer, Michele Sebag, David Silver, Csaba Szepesvári, and Olivier Teytaud. The grand challenge of computer go: Monte carlo tree search and extensions. *Communications of the ACM*, 55(3):106–113, 2012.

[72] Yigal Gerchak, Diwakar Gupta, and Mordechai Henig. Reservation planning for elective surgery under uncertain demand for emergency surgery. *Management Science*, 42(3):321–334, 1996.

[73] Yasin Gocgun and Archis Ghate. Lagrangian relaxation and constraint generation for allocation and advanced scheduling. *Computers & Operations Research*, 39(10):2323–2336, 2012.

[74] Yasin Gocgun and Martin L Puterman. Dynamic scheduling with due dates and time windows: an application to chemotherapy patient appointment booking. *Health care management science*, 17(1):60–76, 2014.

[75] Gagan Goel and Aranyak Mehta. Online budgeted matching in random input models with applications to adwords. In *Proceedings of the nineteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 982–991. Society for Industrial and Applied Mathematics, 2008.

[76] Alexander Goldenshluger and Assaf Zeevi. A linear response bandit problem. *Stochastic Systems*, 3(1):230–261, 2013.

[77] Robert C Griggs, Richard T Moxley, Jerry R Mendell, Gerald M Fenichel, Michael H Brooke, Alan Pestronk, and J Philip Miller. Prednisone in duchenne dystrophy: a randomized, controlled trial defining the time course and dose response. *Archives of neurology*, 48(4):383–388, 1991.

[78] ACCORD Study Group. Effects of intensive blood-pressure control in type 2 diabetes mellitus. *New England Journal of Medicine*, 362(17):1575–1585, 2010.

[79] Diwakar Gupta and Brian Denton. Appointment scheduling in health care: Challenges and opportunities. *IIE transactions*, 40(9):800–819, 2008.

[80] Diwakar Gupta and Lei Wang. Revenue management for a primary-care clinic in the presence of patient choice. *Operations Research*, 56(3):576–592, 2008.

[81] Leslie A Hall, Andreas S Schulz, David B Shmoys, and Joel Wein. Scheduling to minimize average completion time: Off-line and on-line approximation algorithms. *Mathematics of operations research*, 22(3):513–544, 1997.

[82] Nima Hamidi and Mohsen Bayati. A general framework to analyze stochastic linear bandit. *Working Paper, Stanford University, Stanford, CA, arXiv preprint arXiv:2002.05152*, 2020.

[83] W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 04 1970.

[84] Jonathan E Helm, Mariel S Lavieri, Mark P Van Oyen, Joshua D Stein, and David C Musch. Dynamic forecasting and control algorithms of glaucoma progression for clinician decision support. *Operations Research*, 63(5):979–999, 2015.

[85] Jon C Helton and Freddie Joe Davis. Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems. *Reliability Engineering & System Safety*, 81(1):23–69, 2003.

[86] O Hernández-Lerma and JB Lasserre. Error bounds for rolling horizon policies in discrete-time markov control processes. *IEEE Transactions on Automatic Control*, 35(10):1118–1124, 1990.

[87] NH Holford and Karl E Peace. Results and validation of a population pharmacodynamic model for cognitive effects in alzheimer patients treated with tacrine. *Proceedings of the National Academy of Sciences*, 89(23):11471–11475, 1992.

[88] Woonghee Tim Huh, Nan Liu, and Van-Anh Truong. Multiresource allocation scheduling in dynamic environments. *Manufacturing & Service Operations Management*, 15(2):280–291, 2013.

[89] Lesley A Inker, Christopher H Schmid, Hocine Tighiouart, John H Eckfeldt, Harold I Feldman, Tom Greene, John W Kusek, Jane Manzi, Frederick Van Lente, Yaping Lucy Zhang, et al. Estimating glomerular filtration rate from serum creatinine and cystatin c. *New England Journal of Medicine*, 367(1):20–29, 2012.

[90] Patrick Jaillet and Xin Lu. Online stochastic matching: New algorithms with better bounds. *Mathematics of Operations Research*, 39(3):624–646, 2013.

[91] Ruiwei Jiang, Siqian Shen, and Yiling Zhang. Integer programming approaches for appointment scheduling with random no-shows and service durations. *Operations Research*, 65(6):1638–1656, 2017.

[92] Bala Kalyanasundaram and Kirk R Pruhs. An optimal deterministic algorithm for online b-matching. *Theoretical Computer Science*, 233(1-2):319–325, 2000.

[93] Garry Kaplan, Marianne Hamilton Lopez, and J Michael McGinnis. Transforming health care scheduling and access: Getting to now. *Washington DC: Institute of Medicine*, 2015.

[94] Chinmay Karande, Aranyak Mehta, and Pushkar Tripathi. Online bipartite matching with unknown distributions. In *Proceedings of the forty-third annual ACM symposium on Theory of computing*, pages 587–596. ACM, 2011.

[95] Richard M Karp, Umesh V Vazirani, and Vijay V Vazirani. An optimal algorithm for on-line bipartite matching. In *Proceedings of the twenty-second annual ACM symposium on Theory of computing*, pages 352–358. ACM, 1990.

[96] Michal Kaut and Stein W Wallace. Evaluation of scenario-generation methods for stochastic programming, 2003.

[97] Pooyan Kazemian, Jonathan E Helm, Mariel S Lavieri, Joshua D Stein, and Mark P Van Oyen. Dynamic monitoring and control of irreversible chronic diseases with application to glaucoma. *Production and Operations Management*, 28(5):1082–1107, 2019.

[98] Pooyan Kazemian, Mustafa Y Sir, Mark P Van Oyen, Jenna K Lovely, David W Larson, and Kalyan S Pasupathy. Coordinating clinic and surgery appointments to meet access service levels for elective surgery. *Journal of biomedical informatics*, 66:105–115, 2017.

[99] Diwas Singh KC, Stefan Scholtes, and Christian Terwiesch. Empirical research in healthcare operations: past research, present understanding, and future opportunities. *Manufacturing & Service Operations Management*, 22(1):73–83, 2020.

[100] Robert Kleinberg. A multiple-choice secretary algorithm with applications to online auctions. In *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 630–631. Society for Industrial and Applied Mathematics, 2005.

[101] Robert Kleinberg, Aleksandrs Slivkins, and Eli Upfal. Multi-armed bandits in metric spaces. In *Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing*, STOC'08, pages 681–690, New York, NY, USA, 2008. Association for Computing Machinery.

[102] Qingxia Kong, Chung-Yee Lee, Chung-Piaw Teo, and Zhichao Zheng. Scheduling arrivals to a stochastic service delivery system using copositive cones. *Operations research*, 61(3):711–726, 2013.

[103] Panos Kouvelis, Joseph Milner, and Zhili Tian. Clinical trials for new drug development: Optimal investment and application. *Manufacturing & Service Operations Management*, 19(3):437–452, 2017.

[104] Elizabeth F Krakow, Michael Hemmer, Tao Wang, Brent Logan, Mukta Arora, Stephen Spellman, Daniel Couriel, Amin Alousi, Joseph Pidala, Michael Last, et al. Tools for the precision medicine era: how to develop highly personalized treatment recommendations from cohort and registry data using q-learning. *American journal of epidemiology*, 186(2):160–172, 2017.

[105] Eva K Lee, Xin Wei, Fran Baker-Witt, Michael D Wright, and Alexander Quarshie. Outcome-driven personalized treatment design for managing diabetes. *Interfaces*, 48(5):422–435, 2018.

[106] Brian Lemay, Amy Cohn, Marina Epelman, and Stephen Gorga. New methods for resolving conflicting requests with examples from medical residency scheduling. *Production and Operations Management*, 26(9):1778–1793, 2017.

[107] Retsef Levi, Robin O Roundy, and David B Shmoys. Primal-dual algorithms for deterministic inventory problems. *Mathematics of Operations Research*, 31(2):267–284, 2006.

[108] Lihong Li, Yu Lu, and Dengyong Zhou. Provably optimal algorithms for generalized linear contextual bandits. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pages 2071–2080, Sydney, NSW, Australia, 2017. JMLR.org.

[109] Nan Liu, Stacey R Finkelstein, Margaret E Kruk, and David Rosenthal. When waiting to see a doctor is less irritating: Understanding patient preferences and choice behavior in appointment scheduling. *Management Science*, 64(5):1975–1996, 2017.

[110] Nan Liu, Van-Anh Truong, Xinshang Wang, and B Anderson. Integrated scheduling and capacity planning with considerations for patients length-of-stays. To appear in Production and Operations Management, 2019.

[111] Nan Liu, Van-Anh Truong, Xinshang Wang, and Brett R Anderson. Integrated scheduling and capacity planning with considerations for patients length-of-stays. *Production and Operations Management*, 2019.

[112] Nan Liu, Peter van de Ven, and Bo Zhang. Managing appointment booking under customer choices. forthcoming, 2018.

[113] Nan Liu, Peter M van de Ven, and Bo Zhang. Managing appointment booking under customer choices. *Management Science*, 2019.

[114] Nan Liu, Serhan Ziya, and Vidyadhar G Kulkarni. Dynamic scheduling of outpatient appointments under patient no-shows and cancellations. *Manufacturing & Service Operations Management*, 12(2):347–364, 2010.

[115] Alex Macario. What does one minute of operating room time cost? *Journal of clinical anesthesia*, 22(4):233–236, 2010.

[116] James M Magerlein and James B Martin. Surgical demand scheduling: a review. *Health services research*, 13(4):418, 1978.

[117] Mohammad Mahdian and Qiqi Yan. Online bipartite matching with random arrivals: an approach based on strongly factor-revealing lps. In *Proceedings of the forty-third annual ACM symposium on Theory of computing*, pages 597–606. ACM, 2011.

[118] Ho-Yin Mak, Ying Rong, and Jiawei Zhang. Appointment scheduling with limited distributional information. *Management Science*, 61(2):316–334, 2014.

[119] Ho-Yin Mak, Ying Rong, and Jiawei Zhang. Sequencing appointments for service systems using inventory approximations. *Manufacturing & Service Operations Management*, 16(2):251–262, 2014.

[120] Camilo Mancilla and Robert Storer. A sample average approximation approach to stochastic appointment sequencing and scheduling. *IIE Transactions*, 44(8):655–670, 2012.

[121] IHS Markit. The complexities of physician supply and demand: Projections from 2015 to 2030. 2017.

[122] Colin D Mathers and Dejan Loncar. Projections of global mortality and burden of disease from 2002 to 2030. *PLoS medicine*, 3(11):e442, 2006.

[123] Jerrold H May, William E Spangler, David P Strum, and Luis G Vargas. The surgical scheduling problem: Current research and future opportunities. *Production and Operations Management*, 20(3):392–405, 2011.

[124] Glen P Mays, Sharla A Smith, Richard C Ingram, Laura J Racster, Cynthia D Lamberth, and Emma S Lovely. Public health delivery systems: evidence, uncertainty, and emerging research needs. *American Journal of Preventive Medicine*, 36(3):256–265, 2009.

[125] Garth P McCormick. Computability of global solutions to factorable nonconvex programs: Part iconvex underestimating problems. *Mathematical programming*, 10(1):147–175, 1976.

[126] Aranyak Mehta, Amin Saberi, Umesh Vazirani, and Vijay Vazirani. Adwords and generalized on-line matching. In *46th Annual IEEE Symposium on Foundations of Computer Science (FOCS'05)*, pages 264–273. IEEE, 2005.

[127] Yonatan Mintz, Anil Aswani, Philip Kaminsky, Elena Flowers, and Yoshimi Fukuoka. Nonstationary bandits with habituation and recovery dynamics. *Operations Research*, 68(5):1493–1516, 2020.

[128] Douglas J Morrice, Jonathan F Bard, Luci K Leykum, and Susan Noorily. The impact of a patient-centered surgical home implementation on preoperative processes in outpatient surgery. *IISE Transactions on Healthcare Systems Engineering*, 8(2):155–166, 2018.

[129] Rémi Munos et al. From bandits to monte-carlo tree search: The optimistic principle applied to optimization and planning. *Foundations and Trends® in Machine Learning*, 7(1):1–129, 2014.

[130] Susan A Murphy. An experimental design for the development of adaptive treatment strategies. *Statistics in medicine*, 24(10):1455–1481, 2005.

[131] Susan A Murphy, Mark J van der Laan, James M Robins, and Conduct Problems Prevention Research Group. Marginal mean models for dynamic regimes. *Journal of the American Statistical Association*, 96(456):1410–1423, 2001.

[132] Diana M Negoescu, Kostas Bimpikis, Margaret L Brandeau, and Dan A Iancu. Dynamic learning of patient response types: An application to treating chronic diseases. *Management science*, 64(8):3469–3488, 2017.

[133] JP Oudhoff, DRM Timmermans, DL Knol, AB Bijnen, and G Van der Wal. Waiting for elective general surgery: impact on health related quality of life and psychosocial consequences. *BMC Public Health*, 7(1):164, 2007.

[134] Suetonia C Palmer, Dimitris Mavridis, Eliano Navarese, Jonathan C Craig, Marcello Tonelli, Georgia Salanti, Natasha Wiebe, Marinella Ruospo, David C Wheeler, and Giovanni FM Strippoli. Comparative efficacy and safety of blood pressure-lowering agents in adults with diabetes and kidney disease: a network meta-analysis. *The Lancet*, 385(9982):2047–2056, 2015.

[135] Hoda Parvin, Shervin Beygi, Jonathan E Helm, Peter S Larson, and Mark P Van Oyen. Distribution of medication considering information, transshipment, and clustering: Malaria in malawi. *Production and Operations Management*, 27(4):774–797, 2018.

[136] Hoda Parvin, Abhijit Bose, and Mark P Van Oyen. Priority-based routing with strict deadlines and server flexibility under uncertainty. In *Proceedings of the 2009 Winter Simulation Conference (WSC)*, pages 3181–3188. IEEE, 2009.

[137] Jonathan Patrick, Martin L Puterman, and Maurice Queyranne. Dynamic multipriority patient scheduling for a diagnostic resource. *Operations research*, 56(6):1507–1525, 2008.

[138] Trang Pham, Truyen Tran, Dinh Phung, and Svetha Venkatesh. Predicting healthcare trajectories from medical records: A deep learning approach. *Journal of Biomedical Informatics*, 69:218–229, 2017.

[139] Joelle Pineau, Arthur Guez, Robert Vincent, Gabriella Panuccio, and Massimo Avoli. Treating epilepsy via adaptive neurostimulation: a reinforcement learning approach. *International Journal of Neural Systems*, 19(04):227–240, 2009.

[140] Min Qian and Susan A Murphy. Performance guarantees for individualized treatment rules. *Annals of statistics*, 39(2):1180, 2011.

[141] Kent Quanrud and Daniel Khashabi. Online learning with adversarial delays. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, pages 1270–1278, Cambridge, MA, USA, 2015. MIT Press.

[142] Paat Rusmevichientong and John N Tsitsiklis. Linearly parameterized bandits. *Mathematics of Operations Research*, 35(2):395–411, 2010.

[143] Kathleen Russell-Babin and Teri Wurmser. Transforming care through top-of-license practice. *Nursing Management*, 47(5):24–28, 2016.

[144] Daniel Russo and Benjamin Van Roy. Learning to optimize via posterior sampling. *Mathematics of Operations Research*, 39(4):1221–1243, 2014.

[145] Antoine Saure, Jonathan Patrick, Scott Tyldesley, and Martin L Puterman. Dynamic multi-appointment patient scheduling for radiation therapy. *European Journal of Operational Research*, 223(2):573–584, 2012.

[146] Steven M Shechter, Matthew D Bailey, Andrew J Schaefer, and Mark S Roberts. The optimal time to initiate hiv therapy under ordered health states. *Operations Research*, 56(1):20–33, 2008.

[147] Pengyi Shi, Mabel C Chou, JG Dai, Ding Ding, and Joe Sim. Models and insights for hospital inpatient operations: Time-dependent ed boarding time. *Management Science*, 62(1):1–28, 2016.

[148] C Stein, Van-Anh Truong, and Xinshang Wang. Advance reservations with heterogeneous customers. To appear in Management Science, 2019.

[149] Luke Tierney and Joseph B Kadane. Accurate approximations for posterior moments and marginal densities. *Journal of the American Statistical Association*, 81(393):82–86, 1986.

[150] Van-Anh Truong. Optimal advance scheduling. *Management Science*, 61(7):1584–1597, 2015.

[151] Ayten Turkcan, Bo Zeng, and Mark Lawley. Chemotherapy operations planning and scheduling. *IIE Transactions on Healthcare Systems Engineering*, 2(1):31–49, 2012.

[152] Martin J Wainwright. *High-dimensional statistics: A non-asymptotic viewpoint*. Cambridge University Press, Cambridge, United Kingdom, 2019.

[153] Bing Wang, Xingbao Han, Xianxia Zhang, and Shaohua Zhang. Predictive-reactive scheduling for single surgical suite subject to random emergency surgery. *Journal of Combinatorial Optimization*, 30(4):949–966, 2015.

[154] Dongyang Wang, Douglas J Morrice, Kumar Muthuraman, Jonathan F Bard, Luci K Leykum, and Susan H Noorily. Coordinated scheduling for a multi-server network in outpatient pre-operative care. *Production and Operations Management*, 27(3):458–479, 2018.

[155] Tingyan Wang, Robin G Qiu, and Ming Yu. Predictive modeling of the progression of alzheimers disease with recurrent neural networks. *Scientific reports*, 8(1):1–12, 2018.

[156] Xiang Wang, David Sontag, and Fei Wang. Unsupervised learning of disease progression models. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD'14, pages 85–94, New York, NY, USA, 2014. Association for Computing Machinery.

[157] Xinshang Wang and Van-Anh Truong. Multi-priority online scheduling with cancellations. *Operations Research*, 2017.

[158] Xinshang Wang, Van-Anh Truong, and D Bank. Online advance admission scheduling for services, with customer preferences. Working paper, Columbia University, NY, 2015.

[159] Yingfei Wang, Chu Wang, and Warren Powell. The knowledge gradient for sequential decision making with stochastic binary feedbacks. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, pages 1138–1147. JMLR.org, 2016.

[160] Paul K Whelton, Robert M Carey, Wilbert S Aronow, Donald E Casey, Karen J Collins, Cheryl Dennison Himmelfarb, Sondra M DePalma, Samuel Gidding, Kenneth A Jamerson, Daniel W Jones, et al. 2017 acc/aha/aapa/abc/acpm/ags/apha/ash/aspc/nma/pcna guideline for the prevention, detection, evaluation, and management of high blood pressure in adults: a report of the american college of cardiology/american heart association task force on clinical practice guidelines. *Journal of the American College of Cardiology*, 71(19):e127–e248, 2018.

[161] David P Williamson and David B Shmoys. *The design of approximation algorithms*. Cambridge university press, 2011.

[162] Kuang Xu and Carri W Chan. Using future information to reduce waiting times in the emergency department via diversion. *Manufacturing & Service Operations Management*, 18(3):314–331, 2016.

[163] XQ Yang and CJ Goh. A method for convex curve approximation. *European Journal of Operational Research*, 97(1):205–212, 1997.

[164] Jiayu Zhou, Jun Liu, Vaibhav A. Narayan, and Jieping Ye. Modeling disease progression via fused sparse group lasso. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, pages 1095–1103, New York, NY, USA, 2012. Association for Computing Machinery.

[165] Zhengyuan Zhou, Renyuan Xu, and Jose H. Blanchet. Learning in generalized linear contextual bandits with stochastic delays. In *Proceedings of the Advances in Neural Information Processing Systems 32*, NeurIPS'19, pages 5198–5209, Red Hook, NY, USA, 2019. Curran Associates Inc.