Data-Driven Learning and Resource Allocation in Healthcare Operations Management

by

Mohammad Zhalechian

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
2022

Doctoral Committee:

Professor Mark P. Van Oyen, Chair
Assistant Professor Raed Al Kontar
Professor Brian Denton
Associate Professor Cong Shi
Assistant Professor Yuekai Sun

Mohammad Zhalechian
mzhale@umich.edu
ORCID iD: 0000-0002-1174-6102

DEDICATION

To my parents,

for their unconditional love and continuous support.

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my advisor, Professor Mark P. Van Oyen, for his endless support, encouragement, and guidance. I am especially grateful for the ways in which he challenged me to see things from a variety of perspectives. He has been more than an advisor to me. He has taught me how to become a better researcher, teacher, mentor, and how to be a good colleague and collaborator. I appreciate the freedom I was given to make my own path, while enjoying his endless support throughout the tenure of my Ph.D. study.

I would like to express my gratitude to Professor Cong Shi. I have been privileged to work closely under his supervision during my Ph.D. study. He has been always a source of inspiration for me, generous in his time, and truly supportive, especially when needed most. I greatly appreciate the way in which he has shaped my thinking and research direction. When I have encountered barriers, our meetings have been a huge help in lifting my motivation, mood, and creativity. My special gratitude goes to Professor Brian Denton for letting me gain first-hand experience in teaching at our department and for always being supportive of me. I appreciate his enthusiasm for new ideas and eagerness to work with students to promote DEI initiatives. I would like to thank Professor Raed Al Kontar for being a great source of inspiration and support. I was inspired by his passion for teaching and research, and I learned a lot from being a teaching assistant in his class. I am very much thankful to Professor Yuekai Sun for his comments and support for my Ph.D. dissertation as well as my dual Master's in Statistics. I would like to thank Professor Mariel Lavieri and Dr. Joshua Stein for their help and support that have shaped my research. I have greatly benefited from working closely with my colleague and great friend, Esmaeil Keyvanshokooh. I truly enjoyed our journey together. I would also like to thank Arlen Dean for the many hours of discussions we have shared. I am looking forward to our future collaboration and research adventures.

Thank you to my classmates, officemates, and many friends in the department and beyond, too many to list all by name, for their altruistic help and stimulating discussions we had about our research and lives. I would like to thank the collegial faculty members and friendly staff at the University of Michigan, who were always available when I needed help

in research or life.

I have been incredibly lucky to find a group of friends in Michigan to enrich my life. Thanks for creating wonderful memories that I will treasure forever. Our Friday night gatherings have been a highlight of my weeks. The last six months have been an special time period for me, and I truly thank the one who has made this happen by making every breath more enjoyable.

Finally, I want to thank my family for their constant support throughout my life. No achievement in my life would have been possible without them. Words cannot express my gratitude to my parents, Shahnaz and Saeid. Their love, encouragement, and support have been the core of my strength throughout my Ph.D. journey. I owe gratitude to my wonderful siblings, Parisa, Payam, and Peyman. Thanks for always being there for me, even though you are on the other side of the world.

TABLE OF CONTENTS

CHAPTER

LIST OF FIGURES

Figure

LIST OF TABLES

Table

ABSTRACT

The tremendous advances in machine learning and optimization over the past decade have immensely increased the opportunity to personalize and improve decisions for a plethora of problems in healthcare. This brings forward several challenges and opportunities that have been the primary motivation behind this dissertation and its contributions in both practical and theoretical aspects. This dissertation is broadly about sequential decision-making and statistical learning under limited resources. In this area, we treat sequentially arriving individuals, each of which should be assigned to the most appropriate resource. Per each arrival, the decision-maker receives some contextual information, chooses an action, and gains noisy feedback corresponding to the action. The aim is to minimize the regret of choosing sub-optimal actions over a time horizon. We provide data-driven and personalized methodologies for this class of problems. Our data-driven methods adaptively learn from data over time to make efficient and effective real-time decisions for each individual, when resources are limited. With a particular focus on high-impact problems in healthcare, we develop new online algorithms to solve healthcare operations problems. The theoretical contributions lie in the design and analysis of a new class of online learning algorithms for sequential decision-making and proving theoretical performance guarantees for them. The practical contributions are to apply our methodology to solve and provide managerial and practical insights for problems in healthcare, service operations, and operations management in general.

In Chapter II, we study a fundamental problem inherent in many applications, called joint online learning and resource allocation. In a general setting, heterogeneous customers arrive sequentially, each of which can be allocated to a resource in an online fashion. Customers stochastically request resources and the algorithm makes allocations that yield stochastic rewards, which the system receives as feedback outcomes after an uncertain delay. We introduce a generic framework that judiciously synergizes online learning with a broad class of online resource allocation mechanisms. The sequence of customer contexts is adversarial and the customer reward and the resource consumption are stochastic and unknown. First, we propose an online algorithm for a general resource allocation problem, which strikes a three-way balance between exploration, exploitation, and hedging against an adversarial arrival sequence. We provide a performance guarantee for this online algorithm in terms of Bayesian regret. Next, we develop our second online algorithm for an advance scheduling problem and evaluate its theoretical performance. Our second algorithm has a more delicate structure and offers multi-day scheduling while accounting for the no-show behavior of customers. We demonstrate the practicality and efficacy of our methodology using clinical data from a partner health system. Our results show that the proposed algorithms provide promising results compared to several benchmark policies.

In Chapter III, we focus on the choice of care unit type upon admission to the hospital, which is a challenging task due to the wide variety of patient characteristics, uncertain needs of patients, and the limited number of beds in intensive and intermediate care units. The care unit placement decisions involve capturing the trade-off between the benefit of better health outcomes versus the opportunity cost of reserving high-demand beds for potentially more complex patients arriving in the future. By focusing on reducing the readmission risk of patients, we develop an online algorithm under the presence of limited reusable hospital beds. The algorithm is designed to (i) adaptively learn the readmission risk of patients through batch learning with delayed feedback and (ii) choose the best care unit placement for a patient based on the observed contextual information and the occupancy level of the care units. We prove that our online algorithm admits a Bayesian regret bound. We also investigate and assess the effectiveness of our optimization-learning methodology using hospital system data. Our empirical results suggest that implementing our approach provides promising results compared to different benchmark policies and improves the current policy of our partner hospital.

CHAPTER I

Introduction

The unprecedented access to big data coupled with advances in artificial intelligence has immensely increased the opportunity to automate decisions for a wide range of problems. Recently, we have seen a plethora of decision support tools that provide personalized and data-driven decisions for a broad range of applications. In healthcare, decision support tools can provide personalized treatment recommendation based on patients' clinical history and bio-makers. In marketing, such decision support tools have shown the potential to increase revenue by recommending personalized ads tailored to users' demographics and interests. In many real-world problems, it is often the case that decisions should be made sequentially. In particular, users arrive sequentially and a sequence of decisions should be made in real-time. There are two key elements in developing online/sequential decision-making paradigms (1) learning from data and predicting user-specific outcomes for possible decisions and (2) harnessing the predicted outcomes and making personalized decisions in real-time.

In many applications, making good decisions partly depends on learning from good data. A decision-making algorithm with good performance should learn from data and perform well when predicting for cases that do not precisely match the previous observations. The online decision-making paradigm suffers from partial feedback. That is, one can only obtain user outcomes corresponding to the chosen decision and cannot observe the counterfactuals from other decisions that could have been made. In this setting, there is often a lack of diverse and rich data that is available in advance. Adaptive learning is an essential technique to ensure effective and efficient learning. Online decision-making algorithms with adaptive learning provide the opportunity to guide the data gathering process as their decisions affect the future data that will be obtained. Hence, the new information is collected adaptively and the most useful information is obtained as quickly as possible. This leads to a fundamental exploration-exploitation trade-off, where one should exploit the current knowledge to make decisions that increase performance (exploitation) while exploring poorly estimated decisions to achieve better ones in subsequent rounds (exploration).

An underlying issue that appears in many operations research/management (OR/OM) problems is the need to sequentially optimize personalized decisions in the presence of limited resources. Depending on the nature of the problem, the decisions are often limited to either single-use resources or reusable resources. Single-use resources are the ones that can be used only one time (e.g., inventory control and advance scheduling), while reusable resources can be occupied for some time duration and then will be released and become available again (e.g., cloud computing platforms such as Amazon Web Services, rental marketplaces such as Airbnb, and hospital bed management). Surprisingly, most studies in the literature of online decision-making with statistical learning do not consider such critical limitations. This brings forward several challenges and opportunities that have motivated this dissertation and its contributions to both practical and theoretical aspects in the relevant literature.

The unifying theme of this dissertation is designing online decision-making with adaptive learning frameworks under limited resources. Per arrival of a user/patient, the decision-maker receives contextual information, chooses an action that depletes some resources, and gains noisy feedback data corresponding to the user and action. The aim is to minimize the regret of choosing sub-optimal actions over a time horizon. Under the full information assumption (no need for learning), this area has been well-studied in the literature. However, in many real-world applications, it is rare for service providers or clinicians to know an action's reward or service time beforehand. More often, decisions must be made when either no data is available, or a small amount of data is available beforehand. Time and again, we have seen the importance of this need. This is evidenced by several pandemics where various sectors have experienced drastic changes in user/patient behavior and resource consumption, rendering past data unreliable and calling for adaptive learning. This dissertation aims to (i) design easy-to-implement algorithms with a theoretical guarantee for practical problems in healthcare, and (ii) provide effective solutions and insights. It provides data-driven methods building on recent technological advances in optimization, sequential decision-making, and statistical learning theory, as we describe in each chapter. The insights derived can empower healthcare institutions and service industries to deliver personalized and high-quality service.

This dissertation is presented in a multiple manuscript format as independent academic papers. The direct results in Chapters II and III have appeared as individual research papers [111] and [110]. For those interested in methodologies similar to those presented in the dissertation, the author has also significantly contributed to the research manuscripts of [68] and [42].

Chapter II - Online Resource Allocation with Personalized Learning. This chapter was motivated by the expansion of new outpatient space being built for a partner health system and the need for a patient appointment scheduling platform. As the first step, we study a general problem of joint online learning and resource allocation. This is a fundamental problem inherent in many applications. In a general setting, heterogeneous customers arrive sequentially, each of which can be allocated to a resource in an online fashion. Customers stochastically consume the resources, allocation decisions yield stochastic rewards, and the system receives feedback outcomes after an uncertain delay.

Learning under delayed feedback is one of the main complexities of this problem. A common assumption in most online learning settings is that the feedback is received immediately once a decision is made. However, this assumption is not practical in many applications. For example, when a patient is assigned to an appointment, the soonest possible time to receive the feedback of the patient is after the scheduled date of appointment. To deal with having delayed feedback, we introduce an asynchronous strategy for learning. This strategy necessitates new modeling and theoretical innovations to enable designing and evaluating online algorithms. Another major complexity stems from the need for allocating limited resources when there is no information on future arrivals (adversarial arrivals). In particular, we need a robust policy that can be implemented without any information about the evolution of future demands. In this setting, heterogenous users with stochastic resource consumption arrive sequentially over time and the limited resources should be assigned to them to hedge against the future arrival sequence. Inspired by the primal-dual paradigm, we incorporate an online resource allocation mechanism into our algorithms that judiciously allocate the limited resources and hedges against the future arrivals.

In this chapter, we introduce a generic framework that judiciously synergizes online learning with a broad class of online resource allocation mechanisms, where the sequence of customer contexts is adversarial and the customer reward and the resource consumption are stochastic and unknown. We first propose an online algorithm for a general resource allocation problem, which strikes a three-way balance between exploration, exploitation, and hedging against adversarial arrival sequence. We provide a performance guarantee for this online algorithm in terms of regret. Next, we develop our second online algorithm for an advance scheduling problem and evaluate its theoretical performance. Our second algorithm has a more delicate structure and offers multi-day scheduling while accounting for the no-show behavior of customers. The no-show behavior of customers adds a tangled structure to the problem and complicates the learning process. In particular, when a customer is assigned to a server-date pair, the no-show feedback cannot be observed immediately, and customer feedback outcomes cannot be observed at all if the customer does not show up on the service date. To the best of our knowledge, this is the first learning-based advance scheduling algorithm which captures the no-show behavior and has a performance guarantee. We demonstrate the practicality and efficacy of our methodology using clinical data from a partner health system. Our results show that our algorithms provide promising results compared to several benchmark policies.

Chapter III - Personalized Hospital Admission Control: A Contextual Learning Approach. This chapter was motivated by the recent admission control issues at a partner hospital. There are three main categories/types of care units in hospitals, including the intensive care unit (ICU), step down/intermediate care unit (SDU) and general bed unit (GB). The choice of type of care unit for a patient upon admission to the hospital matters and is complicated by the limited unit capacity, high variability in patients' health status, and the high utilization of intensive and intermediate care units. Currently, hospitals rely on available clinical expertise to make care unit placement decisions. However, it is difficult for clinicians to recognize the most appropriate care for a patient with uncertain needs. In hospitals, it might seem good to more generously allocate the higher level of care units to patients. The ICU is the most expensive unit type, followed by the SDU. These high-level care units are scarce relative to the GB. Thus, giving a high-level bed to one patient will often mean that another patient gets denied, because of the limited number of beds offering critical care services. This resource limitation adds the complexity of the need for capturing the trade-off between the (i) benefit of better health outcomes when assigning a patient to a high-level bed and (ii) opportunity cost of reserving high-level beds for potentially more complex patients arriving in the future. In addition, we must estimate the benefit a patient will receive if offered a higher level of care unit.

We modeled this problem as a multi-period admission control with online learning. During each interval, patients arrive sequentially and they must be assigned to the care unit that provides the right type of care. Patient arrivals and lengths of stay are stochastic, and resources (care unit beds) are limited and reusable. By focusing on reducing the readmission risk of patients, we design an online algorithm for care unit placement that aims to minimize the total expected readmission risk over a time horizon. The design of this algorithm necessitates several new contributions to the relevant literature because it lacks online learning algorithms that make sequential decisions under limited reusable resources. On a high level, our algorithm includes two interacting layers: contextual batch learning under delayed feedback and online allocation of reusable resources. Having an accurate belief on the patient outcome (risk of readmission) corresponding to different possible alternatives (care unit placements) is a key ingredient that should be taken into account. Our algorithm adaptively learns the expected risk of readmission using the feedback outcomes observed (whether the patient is readmitted or not). To capture the effect of care unit placement decisions on capacity, we incorporated a policy guide model into our algorithm to approximate the effect of lengths of stay on capacity. Our algorithm judiciously makes care unit placement decisions by leveraging our policy guide model, and the algorithm continuously updates itself using the feedback outcomes.

On the theoretical side, we prove a performance guarantee using the notion of regret. Deriving this regret involves bounding two types of loss: (i) the loss associated with contextual batch learning with delay, and (ii) the loss associated with the allocation of reusable resources. We proposed a new bridging technique to add these losses and derive a bound on the regret. On the practical side, we evaluate the performance of our algorithm using real hospital system data. Our work provides a proof of concept for using our methodology for care unit placements in hospitals. This work provides insights into the potential ability of learning algorithms to reduce readmission rates and possibly other patient outcomes such as mortality risk. Furthermore, our general method can also deliver a cutting-edge methodology to several other applications, including but not limited to computing platforms such as Amazon Web Services (AWS), hospitality services such as Airbnb, and hotel-booking platforms.

Chapter IV - Conclusions and Future Research. This dissertation investigates the long-lasting gap in the area of joint learning and optimization. In Chapter IV, we summarize some of our most important contributions to the healthcare problems of advance scheduling and care unit placements. We also discuss some promising future research avenues that could expand on this dissertation.

CHAPTER II

Online Resource Allocation with Personalized Learning[1]

2.1 Introduction

The rapid growth of information and accessibility to big data provide a unique opportunity to shift toward personalized decision-making. Typically, *personalization* can be achieved by learning from the past data and decision outcomes and making personalized decisions for new users based on their contextual information such as demographic and clinical/web history (see, e.g., [18], [20], and [33]). The growth of accessibility to personalized health information can revolutionize personalization in the healthcare industry. For instance, online appointment platforms have the potential to increase access to high-quality care for all patients by pairing patients with the best available doctors in their area and providing appointments based on patients' conditions and preferences. This generalizes to other areas such as marketing, where personalization can help companies achieve greater click-through rates by offering personalized ads and promotions based on user demographics and interests.

There are two core elements in developing a real-time system for personalized resource allocation, including (i) learning a model that predicts user-specific outcomes for the possible decisions/actions, and (ii) harnessing such a predictive model to make *personalized* resource allocation decisions for subsequent users. This sequential decision-making with learning process suffers from *bandit feedback*, where one can only obtain user reactions/outcomes (often called feedback) for the chosen decision and cannot observe counterfactuals from other decisions that could have been made. This hurdle spurs a more data-efficient method for learning a model rather than offline statistical models. Both offline and online statistical methods rely on historical data to provide estimates; however, online methods offer the

[1] Mohammad Zhalechian, Esmaeil Keyvanshokooh, Cong Shi, and Mark P Van Oyen. Online resource allocation with personalized learning. *Operations Research*, 2022.

advantage of *adaptive learning*. In adaptive learning, new information is collected adaptively such that the most useful information is collected as quickly as possible rather than simply using potentially large historical data. The key to adaptive learning is the *exploration-exploitation* trade-off ([8]). The decision-maker often wishes to exploit the current estimate to make decisions that maximize the reward (exploitation) while exploring more about poorly estimated decisions to achieve higher rewards in subsequent decisions (exploration). The challenge here is to carefully balance this exploration-exploitation trade-off.

In addition to learning, we often must optimize personalized decisions in the presence of *limited* resources and *unknown* future arrivals. In this setting, heterogeneous users with stochastic resource consumption arrive sequentially over time and the limited resources should be assigned to them to hedge against the future arrival sequence. For instance, an appointment scheduling platform needs to schedule upfront appointments (provider-date) to arriving patients in an online fashion under limited availability of providers. Here, not only is the decision-maker often uncertain about the patient rewards and service times, but there is also a lack of information about future sequence of patient types. Thus, patients should be scheduled such that there will be available capacity for high-reward/urgent patients that might show up in the future ([103]).

We consider the problem of online resource allocation with personalized learning and design a real-time system for personalized resource allocation. Customers are often characterized by context vectors that contain their contextual information. In many real-world problems of this type, two important features should be taken into account: (i) the number of future customer arrivals and their context vectors are unknown, and (ii) there are often unknown model parameters vital for making decisions for each arriving customer. Various versions of this problem have been studied extensively in the literature, but mostly *separately* (i.e., either online resource allocation or online learning with limited resources). In the typical online resource allocation problem (adversarial or stochastic), the reward and the resource consumption of each arriving customer are *known* and the difficulty is in conducting resource allocation to hedge against the uncertainty of the future arrival sequence. In the online learning problem with limited resources, the typical assumption is that the customer contexts are independently and identically distributed (IID), which results in the existence of an underlying (fixed) optimal randomized allocation strategy. Thus, the difficulty is to balance the exploration and exploitation trade-off to converge to the optimal allocation strategy. However, *neither* of the models developed for the above-mentioned problems addresses a resource allocation problem with both features (i) and (ii).

Another important but often neglected issue in online learning problems is delayed feedback, which is a key ingredient in the learning process. A common assumption in most online learning settings is that the feedback is *received immediately* once a decision is made; however, it is often realized with delay in many applications. For example, an appointment scheduling platform may schedule hundreds of patients per day while the soonest possible time to receive the feedback of a patient is after the scheduled date of appointment. Furthermore, a realized feedback may not be *processed* immediately after it is received in some cases ([29]), which can be another reason for having delayed feedback.

The above discussions and challenges raise the following two important research questions that we answer in this chapter: (i) how one can provide a generic personalized resource allocation framework that strikes a three-way balance between exploration, exploitation, and hedging against the adversarial arrival sequence under delayed feedback, and (ii) how this framework can be tailored to the advance scheduling problem with multi-day scheduling and no-show behavior.

2.1.1 Contributions and Main Results

We introduce a generic framework that synergizes online contextual learning with a broad class of online resource allocation mechanisms. First, we consider a general resource allocation problem. Next, we consider an advance scheduling problem, which is an application of our generic framework but with a more delicate structure. For each problem, we develop an online algorithm with a theoretical performance guarantee. Below, we shall summarize our main results and contributions.

(a) We develop a new *general Bayesian regret analysis* for online resource allocation algorithms with personalized learning in a setting where the (i) sequence of customer contexts is adversarial (no need for IID assumption), (ii) customer reward is stochastic, and (iii) resource consumption is stochastic. Our analysis allows for seamless integration of competitive ratio bounds for online resource allocation algorithms and Bayesian regret bounds for contextual learning algorithms. In particular, bounding the Bayesian regret of our algorithms necessitates defining an *auxiliary problem* and proposing a set of *bridging techniques* (see proof of Theorem II.1). Note that an auxiliary problem is also defined by [35] to derive a lower bound for resource allocation with learning. However, their auxiliary problem is different from the one required for our analysis (see §2.4.1 for details). By introducing an auxiliary problem in which the unknown model parameters are known but not the sequence of customer contexts, we can decompose the main regret term into two parts: (i) the loss corresponding to uncertainty in rewards and resource consumption values, and (ii) the loss related to the optimality gap of an online resource allocation mechanism for solving the auxiliary problem. Then, we derive the following general performance guarantee:

$$\textsc{BayesReg}(L) \leq \text{CLS-}\textsc{Loss} + (1-\alpha)\,\mathbb{E}[V^{BM}],$$

where $\textsc{BayesReg}(L)$ is the Bayesian regret of the algorithm over the planning horizon of length L (see Definition II.2). The CLS-Loss term comprises the contextual learning loss and the loss associated with the stochastic nature of resource consumption; it is *sub-linear* in the number of arrivals over the planning horizon. Also, α is the competitive ratio of any possible resource allocation mechanism included in the algorithm, V^{BM} is the total expected reward obtained by the benchmark given the model parameters and the sequence of customer contexts, and $\mathbb{E}[V^{BM}]$ is the expected value of V^{BM} over the prior distributions of the model parameters.

(b) Our framework can handle *delayed bandit feedback*. When there is no delay, the updating process for an estimator can be done using a synchronous strategy in which both the context vector and the realized feedback are needed to update the estimator. This strategy does not work without having immediate feedback. Thus, we introduce an *asynchronous* strategy under delayed feedback, which necessitates the development of a new confidence bound. As a by-product of our regret analysis, we develop a confidence bound under delayed feedback for the unknown parameter in a linear model using the ordinary least squares (OLS) method. Assuming $D_{\max}$ as an upper bound on delays, our confidence bound yields a regret bound of $\tilde{\mathcal{O}}\big(d\sqrt{T} + dD_{\max}\big)$ for a standard contextual bandit problem studied by [112] when the reward model is linear. This is a *strictly tighter* bound than the one provided in that study (see §2.4.3 for a detailed comparison), and it is of independent interest beyond this chapter.

(c) In addition to uncertain rewards, there is also uncertainty in *resource consumption* by the heterogeneous customers. Our online algorithms enforce the capacity constraints only in expectation. Accordingly, there is a possibility of exceeding the resource capacity in some cases. We impose a *penalty* on the amount of capacity allocated in excess of the resource capacity and subtract it from the total expected reward of the algorithm obtained by ignoring the possibility of exceeding the resource capacity. We derive a high-probability bound for the penalty term, which is sub-linear in the number of arrivals over the planning horizon (Propositions II.3 and II.4). In the study of [49], an online algorithm is proposed to adaptively learn an unknown parameter in the capacity constraints. For a setting with *finite* contextual information sampled IID from a known distribution, they imposed a penalty term to bound the amount of capacity allocated in excess of the resource capacity and derived an upper bound of $\tilde{\mathcal{O}}\big(\sqrt{|\mathcal{X}|KT}\big)$, where K is the number of actions and $|\mathcal{X}|$ is the number of possible contexts. This bound explodes when the number of possible contexts increases. As a by-product of our algorithm's design and regret analysis techniques, we derive an upper bound of $\tilde{\mathcal{O}}\big((d+K)\sqrt{T}\big)$ for the penalty term in the instantaneous feedback setting, which depends on the *dimension* of the feature space $d+K$ rather than $\sqrt{|\mathcal{X}|K}$.

(d) Our algorithm for advance scheduling not only provides *multi-day* scheduling, but also captures the *no-show* behavior. The bulk of our additional analyses for this setting is related to deriving a confidence bound on the expected reward when there is an unknown probability of no-show. The main source of complexity comes from the fact that when a customer is assigned to a server-date, the no-show feedback cannot be observed immediately, and customer feedback outcomes cannot be observed *at all* if the customer does not show up on the service date. This *tangled structure* for observing feedback outcomes requires deriving a new confidence bound that can be of independent interest in other applications as well (Proposition II.5). To the best of our knowledge, our algorithm is the first learning-based advance scheduling algorithm with a performance guarantee that captures the no-show behavior.

2.1.2 Literature Review

We discuss two major research domains and streams of literature relevant to our work: (i) multi-armed bandits and (ii) online resource allocation.

Multi-armed bandits. Multi-armed bandit (MAB) is an online framework for making sequential decisions over time when there is uncertainty about the effect of each action (arm) on the outcome. In this framework, the agent selects an action from a set of possible actions, then a bandit feedback is revealed that helps the agent make better decisions over time. Contextual MAB is a generalization of MAB in which the reward of each action depends on the observed contextual information at each round. The contextual setting addresses many real-world applications such as online recommendation systems, online advertising, and personalized healthcare. For a comprehensive review of recent MAB studies, we refer to [99].

MAB with Resource Constraints. MAB with resource constraints is a recent class of MAB, where each action consumes a certain amount of the resources. [10] and [4] studied standard MAB with resource constraints and proposed online algorithms with a performance guarantee. In the revenue management area, [49] studied a pricing problem that can be viewed as MAB with resource constraints; they proposed an algorithm to maximize the total revenue with limited inventory. [12], [5], and [3] studied MAB with resource constraints and IID context vectors. [12] extended the general contextual MAB to a resourceful contextual bandit which allows a budget constraint. [5] proposed an efficient algorithm for the contextual bandit with knapsack by generalizing the approach of [2] designed for the non-constraint version of this problem. [3] developed a linear contextual bandit with knapsack and constructed confidence ellipsoids to estimate the unknown parameters. In the above-mentioned studies, the IID assumption for context vectors results in the existence of an underlying optimal randomized allocation strategy. However, there is no (fixed) optimal randomized allocation strategy in our problem setting in which context vectors are picked by an adversary (no IID assumption).

MAB with Delayed Feedback. The issue of delayed feedback has been identified and discussed by the salient empirical study of [29]. [64] studied a stochastic MAB with no side (contextual) information and an adversarial MAB with side information. They showed that the delayed feedback setting causes an additive penalty in the stochastic model and a multiplicative penalty in the adversarial model. [91] studied another MAB with aggregated and anonymous feedback in which only the sum of the previously generated rewards can be observed at the end of each round; they matched the same regret bound as in [64]. [105] analyzed MAB in a setting that user feedback is censored if the delay time exceeds a threshold. [22] studied an adversarial bandit under adversarial delay (arbitrary sequence of delays). They provided a finite-sample delay-adaptive regret bound for the Exp3 algorithm with delay. Contextual bandits with delay are much less explored in the literature. [45] studied a stochastic contextual bandit with fixed delay. For a finite class of policies, they provided a regret bound of $\mathcal{O}(\sqrt{K\log(N)}(\tau_c+\sqrt{T}))$, where N is the number of policies and τ_c is the fixed delay. Recently, [112] studied a contextual generalized linear model (GLM) bandit with delay and proposed an upper confidence bound (UCB) algorithm. When delay is bounded, they established a regret bound of $\tilde{\mathcal{O}}\big((d+\sqrt{dD_{\max}})\sqrt{T}\big)$, where $D_{\max}$ is an upper bound on delays and d is the feature dimension. For the IID setting, they established a regret bound of $\tilde{\mathcal{O}}\big((\sqrt{\mu d}+\sqrt{\sigma d}+d)\sqrt{T}\big)$, where μ and σ are the mean of delay and a parameter to characterize the tail of delay, respectively. Their proof is built upon the analysis provided by [77] for a GLM-UCB algorithm with a warm-up period using a maximum likelihood estimator without regularization. The design of their algorithm and proof ideas are significantly different from those in this chapter, in which we have a linear model and use an OLS estimator with regularization under bounded delayed feedback.

Online Resource Allocation. *Competitive ratio* is the most widely used method for evaluating the performance of online resource allocation algorithms. In particular, it is often defined as the relative performance between an online algorithm and a benchmark (clairvoyant policy) under the worst-case input instance. Since our framework incorporates an online resource allocation mechanism, we also briefly review two relevant streams of literature, including online matching and online advance scheduling.

Online Matching. Studies in this area usually fall into two main settings, including stochastic and adversarial. In the stochastic setting, algorithms either depend heavily on forecasting the arrival pattern using historical data or consider an assumption on the arrival pattern. In the adversarial setting, algorithms do not need an assumption on the arrival pattern and have the key advantage of being robust to possible changes in the arrival pattern. [81] studied the Adwords problem and developed an online algorithm achieving a competitive ratio of $1 - 1/e$ based on a trade-off revealing linear program (LP) technique. [25] developed an elegant primal-dual paradigm for the same problem with the same competitive ratio. Several variants of this problem have been studied in the literature. We refer to [43] and [66] for recent generalizations of this problem. The Bayesian regret of our online algorithms partly depends on the competitive ratio of a resource allocation mechanism used in our algorithms. We introduce two variants of the primal-dual paradigm proposed by [25] and use them as resource allocation mechanisms in our online algorithms. The first one allows customers to have different heterogeneous rewards and resource consumption values. The second version extends the first one to cope with scheduling customers over multiple days.

Online Advance Scheduling. The literature of advance scheduling has focused on two types of waiting, including *direct* and *indirect*. As indicated by [55], direct waiting refers to the time that a customer/patient arrives at the system/clinic on the day of the appointment until the service time; while indirect waiting refers to the time between receiving a request and the actual appointment date. Most of the literature on advance scheduling focuses on intra-day scheduling to reduce direct waiting. Our work focuses on multi-day scheduling to reduce indirect waiting. [90], [56], and [47] proposed several heuristics for this problem and investigated the structural properties of the optimal policies. [103] studied a two-class advance scheduling model and proposed analytical results for this problem. In a stochastic setting with heterogeneous patients, [107] and [101] proposed online algorithms with bounded competitive ratios. For the adversarial setting with heterogeneous patients, [67] developed an online advance scheduling algorithm in which the capacity of providers could be extended at the expense of overtime cost. They provided a competitive ratio using the primal-dual analysis developed by [25]. No-show behavior is another feature of advance scheduling that can be captured in modeling this problem. [78] was the first study that explicitly modeled patient no-shows in an appointment scheduling problem. [47] also studied an appointment scheduling problem with patient no-shows. They characterized the structure of the static optimal policy and provided a bound on the optimality gap in a stochastic setting. To the best of our knowledge, there is no prior research work on advance scheduling with learning and no-show behavior; however, we explicitly model the no-shows in our second model (advance scheduling). It is worth noting that past studies are split on the dependence of the no-show probability on the service/appointment delay (see, e.g., [106] and [78]). Our theoretical performance guarantee for the PAS-LD algorithm holds when the no-show probability is independent of the service delay.

Lastly, the closest works to ours include [89] and [35]. [89] studied an online matching problem with learning, in which several types of patients must be matched to perishable resources, the reward distribution is unknown, and each patient assignment consumes one unit of the capacity. With non-stationary Poisson processes for patient arrivals, they proposed a two-phase algorithm that has distinct phases for exploration and exploitation. They proved that the algorithm achieves a regret bound that increases sub-linearly with the number of planning cycles. [35] studied the problem of allocating limited resources to heterogeneous customers over time, where the rewards are known and the resource consumption distribution associated with each customer type and action should be learned over time. They provided an information-theoretic lower bound without the IID assumption. We provide a *Bayesian regret* for our framework in which (i) we do not have the IID assumption on contexts, so there is no optimal allocation strategy fixed over time, (ii) both reward and resource consumption are stochastic, (iii) the no-show behavior should be taken into account in our second model (advance scheduling), and (iv) the learning process is done under bounded *delayed feedback*.

2.1.3 Organization and General Notation

The remainder of this chapter is organized as follows. We formulate our problem in §2.2 and propose two online algorithms in §2.3. We carry out a non-asymptotic performance analysis in §2.4. In §3.5, we provide a case study using clinical data from a partner health system. Finally, we conclude our chapter in §2.6.

All vectors are column vectors. For any column vector $x \in \mathbb{R}^n$, x' denotes its transpose. The determinant and trace of a square matrix M is denoted by $\det(M)$ and $\operatorname{tr}(M)$, respectively. The Euclidean norm and weighted norm of x are denoted by $\|x\| = \sqrt{x'x}$ and $\|x\|_M = \sqrt{x'Mx}$, respectively. Also, I denotes the identity matrix. For a symmetric positive definite matrix V, we define λ_{min} as the smallest eigenvalue of V. We use $\mathbb{1}(\cdot)$ as the indicator function.

2.2 Personalized Resource Allocation with Learning System Models

In this section, we start by formally defining our framework for a general online resource allocation problem. Then, we tailor our framework to an online advance scheduling problem, which is an application of our generic framework for multi-day scheduling and has a more delicate structure.

We make three assumptions in our models: (i) the sequence of customer contexts is *unknown* (context vectors do not necessarily come from a fixed distribution) and picked by an *oblivious adversary*, which is a non-adaptive adversary that picks the sequence of customer contexts upfront, (ii) the customer-resource (customer-server) match quality with a linear model is stochastic and stationary over time, and it has an *unknown* model parameter that should be learned adaptively, and (iii) the resource consumption (service time) is stochastic and follows an unknown distribution and its mean value may depend on the customer's context. To simplify the presentation and reduce cumbersome notation, we assume that the mean resource consumption by a customer is known upon arrival. Note that this assumption is not necessary and can be dropped via Corollary II.2.

2.2.1 General Online Resource Allocation

Consider a finite planning horizon of length L. We define $\mathcal{L} = \{1, \ldots, L\}$ as the set of days and $\mathcal{K} = \{1, \ldots, K\}$ as the set of resources. For each resource k, there are $\mathcal{C}_k$ available units of capacity that are not replenishable during the considered planning horizon. Let $\mathcal{M}_\ell = \{1, \ldots, M_\ell\}$ be the set of customers who arrive *sequentially* on day $\ell \in \mathcal{L}$. Each customer is associated with a context vector which carries the customer's information. Let $\varphi_i^{\mathcal{X}}(\ell) \in \mathbb{R}^d$ be the *context vector* of the i^{th} customer on day ℓ, and $\varphi_k^{\mathcal{A}}(\ell)$ be a K-dimensional action vector corresponding to the choice of resource k, i.e., $\varphi_k^{\mathcal{A}}(\ell) = (\mathbb{1}_{k=1}, \ldots, \mathbb{1}_{k=K})'$. We define $\phi_{ik}(\ell) = \left(\varphi_i^{\prime\mathcal{X}}(\ell), \varphi_k^{\prime\mathcal{A}}(\ell)\right)'$ as the *feature vector*, which concatenates the context vector $\varphi_i^{\mathcal{X}}(\ell)$ and the action vector $\varphi_k^{\mathcal{A}}(\ell)$. Note that our model can be easily extended to a setting in which any action-context pair can be mapped to a d-dimensional feature vector using a known feature map.

Dynamics. Allocation decisions are made in an online fashion such that customers are handled as they arrive without knowing the future sequence of customer contexts. Upon arrival of the i^{th} customer on day ℓ, a context $\varphi_i^{\mathcal{X}}(\ell)$ is revealed to the system. Then, the customer must be either assigned to an available resource $k \in \mathcal{K}$ or rejected based on the revealed context $\varphi_i^{\mathcal{X}}(\ell)$ and the available observations up to the current time. If this customer is assigned to resource k, it will consume $\mathcal{S}_{ik}(\ell)$ units of capacity in $\mathcal{C}_k$. The dependence on ℓ is only used to pair the customer i to the arrival day ℓ and the amount of resource consumption does not depend on the day of arrival (similarly for the match quality, below). Finally, the system *observes* the reward which depends on the customer-resource match quality $\mathcal{Q}_{ik}(\ell)$ and the resource consumption $\mathcal{S}_{ik}(\ell)$. The match quality feedback is received with delay, which can be an *arbitrary* amount of time. Our performance analysis only requires the existence of an upper bound on the feedback delay. We assume that if there is not enough capacity when a customer is assigned to a resource, the customer still gets fully served but the system only receives a fraction of the reward proportional to the resource consumption not exceeding the available resource capacity.

Modeling reward and resource consumption. Given that a customer is assigned to a resource, the reward depends on two components: (i) customer-resource match quality (per unit of resource consumption) and (ii) resource consumption. The two main components are modeled below.

Customer-resource match quality: The choice of resource $k \in \mathcal{K}$ for the i^{th} customer on day ℓ yields the following stochastic match quality:

$$\mathcal{Q}_{ik}(\ell) = \langle\, \phi_{ik}(\ell), w \rangle + \xi_{ik}(\ell),$$

where $w \in \mathbb{R}^{d+K}$ is the unknown model parameter and $\mathcal{Q}_{ik}(\ell) \in [0, c_Q]$. The noise values, $\xi_{ik}(\ell)$, are independent σ_ξ-sub-Gaussian random variables with zero mean (see Definition II.1).

Definition II.1. A real-valued random variable ξ is σ_ξ-sub-Gaussian if $\mathbb{E}[e^{t\xi}] \le e^{\sigma_\xi^2\, t^2/2}, \quad \forall\, t \in \mathbb{R}$.

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

Resource consumption: Our system model also accounts for stochastic resource consumption values. If resource $k \in \mathcal{K}$ is chosen for the i^{th} customer on day ℓ, the customer uses $\mathcal{S}_{ik}(\ell)$ units of this resource, regardless of ℓ. We assume that $\mathcal{S}_{ik}(\ell)$ is a stochastic resource consumption with known expected value of $s_{ik}(\ell) = \mathbb{E}[\mathcal{S}_{ik}(\ell)] \le c_s$. The noise values, $\eta_{ik}(\ell) = \mathcal{S}_{ik}(\ell) - s_{ik}(\ell)$, are independent σ_η-sub-Gaussian random variables with zero mean. Note that we assume known expected resource consumption (upon arrival), for the sake of reducing cumbersome notation. This assumption is not necessary and can be dropped via Corollary II.2.

Hence, the expected reward of the i^{th} customer on day ℓ assigned to resource k is obtained by:

$$\mathbb{E}[\mathcal{Q}_{ik}(\ell)\ \mathcal{S}_{ik}(\ell)] = \langle\, \phi_{ik}(\ell), w\rangle s_{ik}(\ell),$$

where we assume that the customer-resource match quality is independent of the resource consumption. Although one can argue that the match quality may affect the service time, this independence assumption can be reasonable in many applications. For example, often in practice, hospitals and clinics set the appointment length for a procedure in a manner that does not depend on the provider, which can be captured by our reward model.

The goal in our general online resource allocation problem is to maximize the total expected reward over L days. This can be viewed as maximizing the expected match quality (see, e.g., [85] and [104]) which is weighted by the expected resource consumption (see, e.g., [100] and [81]). For technical reasons, let $\mathcal{H}_{i\ell}$ denote the history available upon the arrival of the i^{th} customer on day ℓ, including context vectors, actions, and realized feedback outcomes. Let $\bar{M}$ be an upper bound on the maximum number of arrivals on each day, and Δ be the maximum number of days required for match quality feedback to be realized. To simplify the notation, we let $q_{ik}(\ell)$ and $r_{ik}(\ell)$ denote $\langle\, \phi_{ik}(\ell), w\rangle$ and $\langle\, \phi_{ik}(\ell), w\rangle s_{ik}(\ell)$, respectively. Let $k(i,\ell)$ be the selected resource for the i^{th} customer on day ℓ, which we will often write it as k^* when indices (i,ℓ) are obvious. Without loss of generality, we assume $\|w\| \leq 1$ and $\|\phi_{ik}(\ell)\| \leq c_\phi$.

2.2.2 Online Advance Scheduling: A More Delicate Model

Online advance scheduling can be viewed as an application of our generic framework. A main *difference* between our general resource allocation and advance scheduling problems is that the latter provides multi-day scheduling and captures the no-show behavior of customers. There is also an additional layer of complexity in the advance scheduling problem with multi-day scheduling regarding the perishable nature of resources. In particular, the remaining capacity can be stored and used later in the general resource allocation problem; however, the capacity (i.e., availability of servers) is perishable in the advance scheduling problem and the remaining availability of servers on a day cannot be transferred to the next day.

Consider a *scheduling horizon* of length L. We define $\mathcal{L} = \{1, \ldots, L\}$ as the set of days, and $\mathcal{K} = \{1, \ldots, K\}$ as the set of servers. Let $\mathcal{M}_\ell = \{1, \ldots, M_\ell\}$ be the set of customers who arrive sequentially on day $\ell \in \mathcal{L}$. Each customer is associated with a context vector (e.g., urgency, request type, and demographics) revealed to the system upon arrival. Let $\varphi_i^{\mathcal{X}}(\ell) \in \mathbb{R}^d$ be the *context vector* of the i^{th} customer on day ℓ. If a customer is accepted, the customer must be assigned to (i) a server $k \in \mathcal{K}$ and (ii) a future service date $t \in \mathcal{L}\backslash\{1\}$. We use both indices $\ell \in \mathcal{L}$ and $t \in \mathcal{L}\backslash\{1\}$ to refer to a day in the scheduling horizon, but index ℓ refers to the arrival day of customers and index t refers to the scheduled date. We define a *server-date* pair (k,t) to refer to the server k and the service date t chosen for a customer. For each day $t \in \mathcal{L}\backslash\{1\}$, each server $k \in \mathcal{K}$ has a limited capacity $\mathcal{C}_{kt}$ which explicitly allows for capacity to vary by day. Let $\varphi^{\mathcal{A}}_{kt}(\ell)$ be an A-dimensional action vector corresponding to the choice of server k and service delay $t-\ell$, which may also include contextual information about the selected server. We define $\phi_{ikt}(\ell) = \left(\varphi'^{\mathcal{X}}_i(\ell), \varphi'^{\mathcal{A}}_{kt}(\ell)\right)'$ as the *feature vector*, which concatenates the context vector $\varphi^{\mathcal{X}}_i(\ell)$ and the action vector $\varphi^{\mathcal{A}}_{kt}(\ell)$.

Dynamics. Upon arrival of the i^{th} customer on day ℓ, a context $\varphi^{\mathcal{X}}_i(\ell)$ is revealed to the system. Then, the customer must be either assigned to an available server-date (k,t) such that $t \geq \ell + 1$ or rejected based on the revealed context vector $\varphi^{\mathcal{X}}_i(\ell)$ and the observations up to the current time. We assume that customers do not cancel the scheduled service, but they may not show up on the service date. If the customer shows up, $\mathcal{S}_{ikt}(\ell)$ units of the server's availability in $\mathcal{C}_{kt}$ is occupied and the system observes two feedback outcomes (i.e., no-show outcome and match quality) with *delay*. If the customer does not show up, the system cannot observe the match quality feedback. The *feedback delay* can be equal to the service delay, but our method allows for additional feedback processing time. To keep the modeling general, we assume that the delay for feedback outcomes can be greater than the service delay but there is an upper bound on it. We assume that if there is not enough capacity when a customer shows up for the scheduled service, the customer still gets fully served but the system only receives a fraction of the reward proportional to the service time not exceeding the server's availability.

Modeling reward and service time. Given that a customer is assigned to a server-date (k,t), the reward depends on three components: (i) no-show outcome, (ii) customer-server match quality (per unit of service time), and (iii) service time. The three main components are modeled below.

No-show: We consider no-show using the binary variable $\mathcal{SU}_i(\ell)$, where $\mathcal{SU}_i(\ell) = 1$ indicates that the i^{th} customer on day ℓ shows up on the service date. We assume that $\mathbb{E}[\mathcal{SU}_i(\ell)] = p$ is *unknown*, where p indicates the probability that the customer shows up on the service date. The noise values, $\epsilon_i(\ell) = \mathcal{SU}_i(\ell) - p$, are independent 1-sub-Gaussian random variables with zero mean.

Note that our model and algorithm are suitable for systems with no cancellation or low cancellation rates. However, they can be easily extended to capture the cancellation behavior when cancellation probabilities are known.

Customer-server match quality: If the system assigns the i^{th} customer on day ℓ to server

$k \in \mathcal{K}$ and the customer shows up, it yields a stochastic match quality $\mathcal{Q}_{ikt}(\ell) = \langle\, \phi_{ikt}(\ell), w\rangle + \xi_{ikt}(\ell)$, where $w \in \mathbb{R}^{d+A}$ is the unknown model parameter and $\mathcal{Q}_{ikt}(\ell) \in [0, c_Q]$. The noise values, $\xi_{ikt}(\ell)$, are independent σ_ξ-sub-Gaussian random variables with zero mean.
Service time: Our framework also accounts for stochastic service times. If server $k \in \mathcal{K}$ is chosen for the i^{th} customer on day ℓ and the customer shows up, it yields a stochastic service time $\mathcal{S}_{ikt}(\ell)$ with the expected value of $s_{ikt}(\ell) = \mathbb{E}\big[\mathcal{S}_{ikt}(\ell)\big] \leq c_s$. The noise values, $\eta_{ikt}(\ell) = \mathcal{S}_{ikt}(\ell) - s_{ikt}(\ell)$, are independent σ_η-sub-Gaussian random variables with zero mean.

Clearly, the system observes the reward if the patient shows up on the service date. Hence, the expected reward of the i^{th} customer on day ℓ assigned to server-date (k, t) can be calculated as:

$$\mathbb{E}\big[\mathcal{SU}_i(\ell)\ \mathcal{Q}_{ikt}(\ell)\ \mathcal{S}_{ikt}(\ell)\big] =\ p\,\langle\, \phi_{ikt}(\ell), w\rangle s_{ikt}(\ell).$$

The goal in our personalized advance scheduling problem is to maximize the total expected reward over a finite scheduling horizon of length L. For technical reasons, let $\bar{\mathcal{H}}_{i\ell}$ denote the history available upon the arrival of the i^{th} customer on day ℓ, including context vectors, actions, and realized feedback outcomes. Let $\bar{M}$ be an upper bound on the maximum number of arrivals on each day, and Δ be the maximum number of days required for a feedback to be realized. Note that we consider the same upper bound Δ for two feedback outcomes only for ease of notation. To simplify the notation, we let $q_{ikt}(\ell)$ and $r_{ikt}(\ell)$ denote $p\,\langle\, \phi_{ikt}(\ell), w\rangle$ and $p\,\langle\, \phi_{ikt}(\ell), w\rangle s_{ikt}(\ell)$, receptively. Let $k(i, \ell)$ and $t(i, \ell)$ be the selected server and service date for the i^{th} customer on day ℓ, respectively. For notational convenience, we will often write them as k^* and t^* when indices (i, ℓ) is obvious. Without loss of generality, we assume $\|w\| \leq 1$ and $\|\phi_{ikt}(\ell)\| \leq c_\phi$.

2.3 Online Algorithms for Resource Allocation and Advance Scheduling

First, we present our online algorithm for a general resource allocation problem, called Personalized Resource Allocation while Learning with Delay (PRA-LD). Next, we present our second online algorithm tailored to the advance scheduling problem, called Personalized Advance Scheduling while Learning with Delay (PAS-LD). In §2.3.1 and §2.3.3, we describe the high-level intuition of our algorithms. We provide the detailed steps of our algorithms in §2.3.2 and §2.3.4.

2.3.1 Main Idea of Personalized Resource Allocation while Learning with Delay

In our general resource allocation problem, customers arrive sequentially over each day, each characterized by a unique set of characteristics (contextual information). Then, each customer must be either assigned to an available resource $k \in \mathcal{K}$ or rejected. Recall that each resource can be used on any day of the planning horizon. If we knew the unknown model parameter w and the sequence of customer contexts $\varphi^{\mathcal{X}} = \{\varphi_i^{\mathcal{X}}(\ell)\}_{\ell \in \mathcal{L}, i \in \mathcal{M}_\ell}$, we could assign customers to resources by solving the following offline optimization model in advance.

$$\max_{x} \quad \sum_{\ell=1}^{L} \sum_{i=1}^{M_\ell} \sum_{k=1}^{K} r_{ik}(\ell)\, x_{ik}(\ell) \tag{2.1}$$

$$\text{s.t.} \quad \sum_{\ell=1}^{L} \sum_{i=1}^{M_\ell} s_{ik}(\ell)\, x_{ik}(\ell) \;\leq\; \mathcal{C}_k, \quad \forall\, k \in \mathcal{K} \tag{2.2}$$

$$\sum_{k=1}^{K} x_{ik}(\ell) \;\leq\; 1, \quad \forall\, \ell \in \mathcal{L}, \;\; \forall\, i \in \mathcal{M}_\ell \tag{2.3}$$

$$x_{ik}(\ell) \;\in\; \{0, 1\}, \quad \forall\, \ell \in \mathcal{L}, \;\; \forall\, i \in \mathcal{M}_\ell, \;\; \forall\, k \in \mathcal{K}, \tag{2.4}$$

where $x_{ik}(\ell)$ is corresponding to assigning the i-th customer on day ℓ to resource k. The objective function (2.1) is to maximize the total expected reward. Constraint (2.2) ensures that the sum of expected resource consumption values by the customers assigned to each resource does not exceed the capacity. Constraint (2.3) ensures that each customer is not assigned to more than one resource.

However, we know neither the unknown model parameter w associated with match quality nor the sequence of customer contexts in advance. In fact, we need to *learn* the distribution of the unknown model parameter as customers arrive *sequentially* over each day. Hence, we design an online algorithm that simultaneously learns the distribution of the unknown model parameter and makes online resource allocation decisions judiciously without knowing the future sequence of customer contexts, which are chosen adversarially. The PRA-LD algorithm leverages contextual learning and online resource allocation techniques to overcome these hurdles and provides personalized decisions.

The PRA-LD algorithm has four main steps. Through the first and second steps, the algorithm provides an individualized estimate for the expected reward of each arriving customer. This estimate depends on the unknown model parameter that should be learned iteratively based on the available observations up to the current time. Here, the exploration-exploitation trade-off is a major challenge. For instance, based on our uncertain estimates in the early days of the planning horizon, we may incorrectly conclude that a specific resource is not a good choice for a customer with certain characteristics, and consequently, we may not be able to identify this incorrect belief without making a different allocation decision for a very similar customer. Inspired by the idea of *posterior sampling* (PS), our algorithm computes a posterior distribution for the unknown model parameter and then takes random samples from the posterior distribution. The intuition behind this sampling is to provide the opportunity to explore alternative choices (carry out exploration) and balance the *exploration-exploitation trade-off*. In our algorithm, we assume a Gaussian prior distribution over the unknown model parameter but our theoretical results are prior-independent.

In the third step, the algorithm either assigns a resource to the arriving customer or rejects the customer according to the available capacity of the resources, the expected resource consumption, and the sample reward. The assignment decision should be made under *no assumption* on the future sequence of customer contexts. If a policy myopically offers the best possible assignment to each customer (i.e., greedy policy), it will not produce a robust solution in some cases. For instance, consider a scenario where a system mostly receives low-reward customers in the first half of the planning horizon while it receives high-reward customers in the second half. In this scenario, most available resources may be occupied by the low-reward customers who arrived in the first half of the planning horizon. Thus, the high-reward customers would be either rejected or assigned to sub-optimal resources because of the limited capacity.

To overcome this problem, we employ a mechanism to make online resource allocation decisions that has good performance compared to the decisions that could have been generated by a natural LP-relaxation of the optimization model (2.1) − (2.4). Thus, our framework not only learns the distribution of the unknown model parameter, but also avoids greedy resource allocation by adopting a resource allocation mechanism.

In the final step, the algorithm leverages the newly realized match quality feedback outcomes to update the posterior distribution of the unknown model parameter. The feedback is revealed with some *delay* after assigning a customer to a resource. The estimator gets updated on the fly following our asynchronous strategy. That is, we update the estimator after assigning each customer to a resource using the realized feature vector, but the customer feedback is included in the updating process once it gets realized.

2.3.2 Personalized Resource Allocation while Learning with Delay

Let (ℓ, i, k^*) be a tuple referring to the i^{th} customer on day ℓ assigned to resource k^*. To update the posterior distribution over vector w, we define $\mathcal{F}^{(\mathcal{Q})}(i, \ell)$ as a set that contains tuples (s, n, k^*) of customers with realized match quality feedback outcomes after the last update, where $s \leq \ell$ and $n < i$. We provide the detailed steps of PRA-LD in Algorithm 1.

Algorithm 1 PRA-LD Algorithm

```
1:  Initialize $m_1^1$ and $B_1^1$ as the mean and the covariance matrix of the Gaussian prior over vector $w$, receptively.
2:  for $\ell = \{1, \cdots, L\}$ do
3:      for each arriving customer $i \in \mathcal{M}_\ell$ on day $\ell$ do
4:          Observe the contextual information $\varphi_i^{\mathcal{X}}(\ell)$ of the customer.
5:          Sample $\tilde{w}_i(\ell)$ from $\mathcal{N}(m_i^\ell, (B_i^\ell)^{-1})$.
6:          Set $\tilde{q}_{ik}(\ell) = \langle \phi_{ik}(\ell), \tilde{w}_i(\ell) \rangle$, and $s_{ik}(\ell) = \mathbb{E}[\mathcal{S}_{ik}(\ell)]$, $\forall\, k \in \mathcal{K}$.
7:          Assign the customer to a resource or reject the customer following $x_{ik}(\ell)$ for $k \in \mathcal{K}$ obtained by a resource allocation mechanism $O^{PRA}\big(\tilde{r}_{ik}(\ell), s_{ik}(\ell)\big)$, where $\tilde{r}_{ik}(\ell) = \tilde{q}_{ik}(\ell) s_{ik}(\ell)$.
8:          Obtain set $\mathcal{F}^{(\mathcal{Q})}(i, \ell)$.
9:          if $i \neq M_\ell$ then
10:             Set $B_{i+1}^\ell = B_i^\ell + \phi_{ik^*}(\ell)\ \phi_{ik^*}^{'}(\ell)$.
11:             Set $g_{i+1}^\ell = g_i^\ell + \sum_{(s,n,k^*) \in \mathcal{F}^{(\mathcal{Q})}(i,\ell)} \mathcal{Q}_{nk^*}(s)\ \phi_{nk^*}(s)$.
12:             Set $m_{i+1}^\ell = (B_{i+1}^\ell)^{-1}\ g_{i+1}^\ell$.
13:         else:
14:             Repeat Steps (10-12) by replacing $\Pi_{i+1}^\ell$ with $\Pi_1^{\ell+1}$, where $\Pi \in \{B, g, m\}$.
```

Description. At first, the prior parameters are initialized based on prior beliefs. There are four main steps in each of the L days. First, upon arrival of a customer, a random sample is drawn from the posterior distribution of w. Next, a sample reward is obtained using the random sample obtained in the first step and the realized context vector. The expected resource consumption, which can depend on the customer's context, is also obtained as it is known in our setting. Then, each arriving customer is either assigned to a resource or rejected following the assignment decision $x_{ik}(\ell)$ obtained by a general resource allocation mechanism $O^{PRA}\big(\tilde{r}_{ik}(\ell), s_{ik}(\ell)\big)$, where $x_{ik}(\ell) = 1$ indicates that the i^{th} customer on day ℓ must be assigned to resource k. Finally, we update the posterior distribution of w after assigning a customer to a resource following Steps (8-14). Note that we do not use the context vectors of rejected customers for updating the estimators. Our updating process is *on the fly* such that the estimator uses the realized feature vector of the customer assigned to a resource, but the feedback of a customer is later included in the updating process when it gets realized. The updating equations follow a Bayesian inference procedure to update the posterior distribution (see [6] for details).

Resource Allocation Mechanism. While the PRA-LD algorithm is designed to work with many resource allocation mechanisms, here we focus on Mechanism 1 as an example for the resource allocation mechanism in Step 7 of Algorithm 1. In particular, we adapt the primal-dual paradigm proposed by [25] for our setting. The *key point* of this primal-dual paradigm is that it maintains a set of dual variables to guide the primal solution, and the evolution of the primal solution determines the update of the dual variables. Consider the dual problem of a natural LP-relaxation of the optimization model (2.1 − 2.4):

$$\begin{aligned} \min_{y,\theta} \quad & \sum_{k=1}^{K} y_k\, \mathcal{C}_k + \sum_{\ell=1}^{L} \sum_{i=1}^{M_\ell} \theta_i(\ell) \\ \text{s.t.} \quad & s_{ik}(\ell)\, y_k + \theta_i(\ell) \ \geq \ r_{ik}(\ell), \quad \forall\, \ell \in \mathcal{L}, \ \forall\, i \in \mathcal{M}_\ell, \ \forall\, k \in \mathcal{K} \\ & y_k, \ \theta_i(\ell) \ \geq \ 0, \quad \forall\, \ell \in \mathcal{L}, \ \forall\, i \in \mathcal{M}_\ell, \ \forall\, k \in \mathcal{K}, \end{aligned}$$

where y_k and $\theta_i(\ell)$ are dual variables corresponding to constraints (2.2) and (2.3), respectively.

Mechanism 1 Primal-Dual Resource Allocation Mechanism

```
1: Set $k^* = \arg\max_{k \in \mathcal{K}} \{\tilde{r}_{ik}(\ell) - s_{ik}(\ell)\, y_k\}$.
2: if $(\tilde{r}_{ik^*}(\ell) - s_{ik^*}(\ell)\, y_{k^*}) \geq 0$ then
3:     Set $x_{ik^*}(\ell) \ = \ 1$ .
4:     Set $\theta_i(\ell) \ = \ \tilde{r}_{ik^*}(\ell) - s_{ik^*}(\ell)\, y_{k^*}$.
5:     Set $y_{k^*} \ = \ y_{k^*}\left(1 + \frac{s_{ik^*}(\ell)}{\mathcal{C}_{k^*}}\right) \ + \ \beta\left(\frac{\tilde{r}_{ik^*}(\ell)}{\mathcal{C}_{k^*}}\right)$.
6: else:
7:     Set $x_{ik}(\ell) \ = \ 0, \ \forall\, k \in \mathcal{K}$.
```

In Mechanism 1, the dual variables y_k and $\theta_i(\ell)$ are initially set to zero for $\ell \in \mathcal{L}, i \in \mathcal{M}_\ell, \ k \in \mathcal{K}$. Upon arrival of a customer, we find a candidate resource k^* that maximizes the term in Step 1, which is the sample reward minus the cost of allocating the expected capacity. This term can be viewed as the *acceptance/rejection criterion* such that a positive value for the candidate resource k^* will assign the customer to this resource; otherwise, the customer should be rejected. If we assign the customer to resource k^*, we update the dual variable y_{k^*} in an incrementally increasing fashion in Step 5. The first term on the right-hand side of Step 5 increases y_{k^*} by an amount proportional to the fraction of the capacity used by the accepted customer. The second term depends on β and the ratio of the estimated reward to the total capacity. The multiplicative updating equation in Step 5 must ensure that there is a sufficiently large increase in y_{k^*} to prevent acceptance of any future customer when the capacity is exhausted in expectation. To achieve this, we set β to $\frac{\eta_{\max}}{\Gamma-1}$, where $\eta_{\max} = \max_{i,k,\ell}\{\tilde{q}_{ik}(\ell)\}$, $\Lambda = \max_{i,k,\ell}\left(\frac{s_{ik}(\ell)}{\mathcal{C}_k}\right)$, and $\Gamma \ = \ (1+\Lambda)^{1/\Lambda}$. Note that setting the value of β requires knowing the values of $\eta_{\max}$ and Λ. Although the sample match quality and the expected resource consumption values (if the resource consumption depends on the customer's context) are unknown in advance, $\eta_{\max}$ and Λ can be calculated in many applications because the range of the match quality and the resource consumption values are often chosen by decision makers and known a priori. The updating equation for dual variable $\theta_i(\ell)$ in Step 4 ensures that the dual problem remains feasible.

2.3.3 Main Idea of Personalized Advance Scheduling while Learning with Delay

In our advance scheduling problem, upon arrival, a customer should be either rejected or assigned to a server and a future service date within the scheduling horizon. If we knew the unknown model parameters (p and w) and the sequence of customer contexts $\varphi^{\mathcal{X}} = \{\varphi_i^{\mathcal{X}}(\ell)\}_{\ell\in\mathcal{L}, i\in\mathcal{M}_\ell}$, we could schedule customers by solving the following offline optimization model at the beginning of the scheduling horizon.

$$\max_x \quad \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\sum_{t=\ell+1}^{L} r_{ikt}(\ell)\, x_{ikt}(\ell) \tag{2.5}$$

$$\text{s.t.} \quad \sum_{\ell=1}^{t-1}\sum_{i=1}^{M_\ell} s_{ikt}(\ell)\, x_{ikt}(\ell) \;\leq\; \mathcal{C}_{kt}, \quad \forall\, k\in\mathcal{K},\; \forall\, t\in\mathcal{L}\backslash\{1\} \tag{2.6}$$

$$\sum_{k=1}^{K}\sum_{t=\ell+1}^{L} x_{ikt}(\ell) \;\leq\; 1, \quad \forall\, \ell\in\mathcal{L}, \quad \forall\, i\in\mathcal{M}_\ell \tag{2.7}$$

$$x_{ikt}(\ell) \;\in\; \{0,1\}, \quad \forall\, \ell\in\mathcal{L},\; \forall\, i\in\mathcal{M}_\ell,\; \forall\, k\in\mathcal{K},\; \forall\, t\geq \ell+1, \tag{2.8}$$

where $x_{ikt}(\ell)$ is corresponding to assigning the i^{th} customer on day ℓ to server-date (k,t). The objective function (2.5) is to maximize the total expected reward. Constraint (2.6) ensures that the sum of the expected service times of the customers assigned to a server on a day does not exceed the availability of the server on that day. Constraint (2.7) ensures that each customer is not assigned to more than one server-date. Note that the scheduled service date for an arriving customer must be after the arrival day (i.e., t should be equal or greater than $\ell+1$).

In our online setting, solving the above optimization model is not possible because the model parameters and the sequence of customer contexts are unknown. Unlike Algorithm 1, the PAS-LD algorithm is designed to allow for the possibility of no-show and multi-day scheduling. The proposed algorithm has four main steps. Through the first and second steps, the algorithm provides customer-specific estimates necessary for the advance scheduling. The reward used in the algorithm depends on the quality of match and the probability of no-show. To carry out exploration, the algorithm takes random samples from the posterior distribution over pw and uses them to obtain sample rewards. It also obtains the expected service time as it is known in our setting.

In the third step, the algorithm either assigns an arriving customer to a server-date (multi-day scheduling) or rejects the customer in an online fashion when there is *no assumption* on the future sequence of customer contexts. Similar to Algorithm 1, any advance scheduling mechanisms appropriate for our setting can be incorporated.

In the final step, the algorithm leverages the realized quality of match and no-show feedback outcomes to update the posterior distribution. If a customer shows up on the service date, both feedback outcomes will be revealed with delay; otherwise, only the no-show feedback will be revealed. We follow our asynchronous strategy for updating the estimator.

2.3.4 Personalized Advance Scheduling while Learning with Delay

Let (ℓ, i, k^*, t^*) be a tuple referring to the i^{th} customer on day ℓ assigned to server-date (k^*, t^*). To update the posterior distribution over vector pw, we define $\mathcal{F}^{(\mathcal{SU},\mathcal{Q})}(i,\ell)$ as a set that contains tuples (s, n, k^*, t^*) of customers with realized no-show and match quality feedback outcomes after the last update, where $s \leq \ell$ and $n < i$. We provide the detailed steps of PAS-LD in Algorithm 2.

Algorithm 2 PAS-LD Algorithm

```
1:  Initialize $m_1^1$ and $(B_1^1)^{-1}$ as the mean vector and the covariance matrix of the Gaussian
    prior over vector $pw$, respectively.
2:  for $\ell = \{1, \cdots, L\}$ do
3:      for each arriving customer $i \in \mathcal{M}_\ell$ on day $\ell$ do
4:          Observe the contextual information $\varphi_i^{\mathcal{X}}(\ell)$ of the customer.
5:          Sample $\tilde{w}_i^c(\ell)$ from $\mathcal{N}(m_i^\ell, (B_i^\ell)^{-1})$.
6:          Set $\tilde{q}_{ikt}(\ell) = \langle \phi_{ikt}(\ell), \tilde{w}_i^c(\ell) \rangle$, and $s_{ikt}(\ell) = \mathbb{E}[\mathcal{S}_{ikt}(\ell)]$,   $\forall\, k \in \mathcal{K},\ \forall\, t \geq \ell + 1$.
7:          Assign the customer to a server-date or reject following $x_{ikt}(\ell)$ for
    $k \in \mathcal{K},\ t \geq \ell + 1$ obtained by an advance scheduling mechanism $O^{PAS}\big(\tilde{r}_{ikt}(\ell), s_{ikt}(\ell)\big)$,
    where $\tilde{r}_{ikt}(\ell) = \tilde{q}_{ikt}(\ell) s_{ikt}(\ell)$.
8:          Obtain set $\mathcal{F}^{(\mathcal{SU},\mathcal{Q})}(i,\ell)$.
9:          if $i \neq M_\ell$ then
10:             Set $B_{i+1}^\ell = B_i^\ell + \phi_{ik^*t^*}(\ell)\ \phi_{ik^*t^*}^{'}(\ell)$.
11:             Set $g_{i+1}^\ell = g_i^\ell + \sum_{(s,n,k^*,t^*) \in \mathcal{F}^{(\mathcal{SU},\mathcal{Q})}(i,\ell)} \mathcal{SU}_n(s) \mathcal{Q}_{nk^*t^*}(s)\ \phi_{nk^*t^*}(s)$.
12:             Set $m_{i+1}^\ell = (B_{i+1}^\ell)^{-1}\ g_{i+1}^\ell$.
13:         else:
14:             Repeat Steps (10-12) by replacing $\Pi_{i+1}^\ell$ with $\Pi_1^{\ell+1}$, where $\Pi \in \{B, g, m\}$.
```

Description. The prior parameters can be initialized based on prior beliefs. The algorithm proceeds over L days. Unlike Algorithm 1, sample rewards should be obtained by considering the probability of no-show. The decision variables $x_{ikt}(\ell)$ are obtained by a general advance scheduling mechanism $O^{PAS}\big(\tilde{r}_{ikt}(\ell), s_{ikt}(\ell)\big)$, where $x_{ikt}(\ell) = 1$ indicates that the i^{th} customer on day ℓ must be assigned to server-date (k, t). The estimators are updated following Steps (8-14) in the algorithm.

Advance Scheduling Mechanism. As an example for the advance scheduling mechanism in Step 7 of Algorithm 2, we focus on Mechanism 2 which is a modified version of the Mechanism 1 tailored to *multi-day* scheduling. This mechanism is designed based on the primal-dual paradigm which maintains a set of dual variables to guide the primal solution. Consider the dual problem for a natural LP-relaxation of the optimization model (2.5 − 2.8):

$$\begin{aligned}
\min_{y,\theta} \quad & \sum_{k=1}^{K}\sum_{t=2}^{L} y_{kt}\, \mathcal{C}_{kt} + \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \theta_i(\ell) \\
\text{s.t.} \quad & s_{ikt}(\ell)\, y_{kt} + \theta_i(\ell) \;\geq\; r_{ikt}(\ell), \qquad \forall\, \ell \in \mathcal{L},\;\; \forall\, i \in \mathcal{M}_\ell,\;\; \forall\, k \in \mathcal{K},\;\; \forall\, t \geq \ell+1 \\
& y_{kt},\; \theta_i(\ell) \;\geq\; 0, \qquad \forall\, \ell \in \mathcal{L},\;\; \forall\, i \in \mathcal{M}_\ell,\;\; \forall\, k \in \mathcal{K},\;\; \forall\, t \geq \ell+1,
\end{aligned}$$

where y_{kt} and $\theta_i(\ell)$ are dual variables corresponding to constraints (2.6) and (2.7), respectively.

In Mechanism 2, the dual variables y_{kt} and $\theta_i(\ell)$ are initially set to zero for $\ell \in \mathcal{L}, i \in \mathcal{M}_\ell,\ k \in \mathcal{K},\ t \geq \ell+1$. The intuition is similar to Mechanism 1. Upon arrival of a customer, we find a candidate server-date (k^*, t^*) that maximizes the term in Step 1. Note that we only search over $t \geq \ell + 1$ to ensure that the candidate service date for the customer is after the arrival day. The term in Step 2 can be viewed as the *acceptance/rejection criterion* such that a positive value for the candidate (k^*, t^*) will assign the customer to this server-date; otherwise, the customer should be rejected. If we assign the customer to server-date (k^*, t^*), we update the dual variable $y_{k^*t^*}$ in an incrementally increasing fashion in Step 5 to make sure that the capacity constraints hold in expectation. We set $\widetilde{\beta} = \frac{\widetilde{\eta}_{\max}}{(\widetilde{\Gamma}-1)}$, where $\widetilde{\eta}_{\max} = \max\limits_{i,k,t,\ell}\big\{\tilde{q}_{ikt}(\ell)\big\}$, $\widetilde{\Lambda} = \max\limits_{i,k,t,\ell}\big(\frac{s_{ikt}(\ell)}{\mathcal{C}_{kt}}\big)$, and $\widetilde{\Gamma} = (1 + \widetilde{\Lambda})^{1/\widetilde{\Lambda}}$. Note that the updating equation for the dual variable $\theta_i(\ell)$ in Step 4 ensures that the dual problem remains feasible.

Mechanism 2 Primal-Dual Advance Scheduling Mechanism

```
1: Set $(k^*, t^*) = \arg\max_{k \in \mathcal{K},\, t \geq \ell+1} \big\{\tilde{r}_{ikt}(\ell) - s_{ikt}(\ell)\, y_{kt}\big\}$.
2: if $(\tilde{r}_{ik^*t^*}(\ell) - s_{ik^*t^*}(\ell)\, y_{k^*t^*}) \geq 0$ then
3:     Set $x_{ik^*t^*}(\ell) \;=\; 1$ .
4:     Set $\theta_i(\ell) \;=\; \tilde{r}_{ik^*t^*}(\ell) - s_{ik^*t^*}(\ell)\, y_{k^*t^*}$.
5:     Set $y_{k^*t^*} \;=\; y_{k^*t^*}\Big(1 + \frac{s_{ik^*t^*}(\ell)}{\mathcal{C}_{k^*t^*}}\Big) \;+\; \widetilde{\beta}\,\Big(\frac{\tilde{r}_{ik^*t^*}(\ell)}{\mathcal{C}_{k^*t^*}}\Big)$.
6: else:
7:     Set $x_{ikt}(\ell) \;=\; 0,\;\; \forall\, k \in \mathcal{K},\; \forall\, t \geq \ell+1$.
```

2.4 Theoretical Performance Analysis and Discussions

We derive a non-asymptotic (i.e., finite-time) performance guarantee for the PRA-LD and PAS-LD algorithms using the notion of Bayesian regret. We start by discussing the performance measure and the auxiliary problem in §2.4.1. We then define the benchmarks and state the main theoretical results in §2.4.2. Finally, we discuss our results and position them in the related literature in §2.4.3.

2.4.1 Performance Measure and Auxiliary Problem

Performance Measure. We evaluate the performance of our algorithms in terms of the *Bayesian regret*, which is a standard metric in the literature (see, e.g., [94] and [95]). This metric is called Bayesian regret since it represents the Bayes risk. The algorithm's regret measures the cumulative loss relative to a benchmark, and the algorithm's Bayesian regret is simply the expected regret over the prior distribution of the unknown model parameter. Bayesian regret has two main advantages: (i) it allows for an arbitrary prior distribution over the unknown model parameter, and (ii) it makes a connection between the PS-based and the UCB-based methods, which provides the opportunity to leverage some of the appealing theoretical properties of the UCB-based methods in deriving the Bayesian regret.

Definition II.2 (Regret and Bayesian Regret). Given the unknown model parameter ϑ, the regret over the planning horizon of length L is defined by

$$\textsc{Reg}(L, \vartheta) = \mathbb{E}\big[OFV^{\pi} - OFV^{ALG} | \vartheta\big],$$

where OFV^{π} and OFV^{ALG} are the total rewards obtained by the optimal policy and the online algorithm, respectively. The conditional expectation is taken over the random realizations given ϑ (e.g., realizations of rewards and resource consumption values), and possible randomization in the online algorithm (e.g., random samples).

Bayesian regret over the planning horizon of length L is then defined by

$$\textsc{BayesReg}(L) = \mathbb{E}\big[\textsc{Reg}(L, \vartheta)\big],$$

where the expectation is taken over the prior distribution of ϑ.

Auxiliary Problem. Defining an auxiliary problem and introducing a bridging technique, we provide a general approach for analyzing the performance of online resource allocation with personalized learning algorithms. This allows for seamless integration of competitive ratio bounds for online resource allocation algorithms and Bayesian regret bounds for contextual learning algorithms. The auxiliary problem can be *formally* defined as an online resource allocation problem, where customers arrive sequentially and the available resources should be allocated to hedge against the adversarial arrival sequence of customers in the future. The main *difference* between the original problem and the auxiliary problem is that the model parameters are *known* to the latter, unlike the former. It is worth noting that the notion of auxiliary problem is introduced in the recent study of [35] to derive an information theoretic lower bound for resource allocation with learning. However, the auxiliary problem defined by [35] is different from the one we defined and used in our analysis. Their auxiliary problem only focuses on the exploration-exploitation trade-off, assuming no resource constraints. Thus, it can be viewed as a contextual bandit problem. In contrast, our auxiliary problem only focuses on the online resource allocation, assuming known model parameters. Since we are working with different auxiliary problems, a different bridging technique is required to derive the Bayesian regret of our algorithms.

We define two auxiliary problems corresponding to the general resource allocation and advance scheduling problems. In the first auxiliary problem, upon arrival of a customer, the expected match quality and the expected resource consumption of the customer are known. Then, a decision (either assigning the customer to a resource or rejecting the customer) should be made using a mechanism which guarantees that the capacity constraints hold in expectation. The second auxiliary problem corresponding to our advance scheduling problem can be defined similarly such that the expected match quality, the probability of no-show, and the expected service time are known upon arrival of a customer in this problem. To keep the results general in our theorems, we derive theoretical performance guarantees for our algorithms with general mechanisms; but the corollaries provide performance guarantees with respect to specific mechanisms.

2.4.2 Performance of Proposed Online Algorithms

First, we define a benchmark for the PRA-LD algorithm in §2.4.2.1, and provide a roadmap for proving its main theoretical result (Theorem II.1) in §2.4.2.2. Next, we define another benchmark for the PAS-LD algorithm in §2.4.2.3, and provide a roadmap for proving its main theoretical result (Theorem II.2) in §2.4.2.4. We also provide theoretical results for PRA-LD and PAS-LD when adapting primal-dual resource allocation mechanisms in Corollaries II.1 and II.3.

2.4.2.1 Benchmark for the PRA-LD Algorithm

We need to define a benchmark to evaluate the performance of our PRA-LD algorithm. Since our general resource allocation problem incorporates both online learning and online resource allocation tasks, the benchmark should be clairvoyant in terms of both tasks. An *offline optimal* benchmark that knows the unknown model parameter w, the sequence of customer contexts, and the realizations of rewards and resource consumption values in advance is too strong to compete with. An alternative is the optimal online *clairvoyant* policy that knows the model parameter and the sequence of customer contexts, but not the realizations of rewards and resource consumption values. Instead, we introduce a *stronger offline* LP-based benchmark that has access to the same information as the clairvoyant policy, but only needs to satisfy the capacity constraints in *expectation* as opposed to all realizations. In particular, our offline LP-based benchmark is a relaxation of the optimal online clairvoyant policy by only requiring the capacity constraints to hold in expectation. Offline LP-based benchmarks are commonly used in recent literature of online resource allocation problems (e.g., [53], [54], and [48]). Our offline LP-based benchmark has two main advantages: (i) as we will show, it provides an *upper bound* on the expected total reward of the clairvoyant policy, and (ii) it naturally provides the opportunity to incorporate primal-dual-based resource allocation mechanisms in our online algorithm.

To evaluate the performance of the PRA-LD algorithm, we formulate the offline LP-based benchmark-I which knows the unknown model parameter w and the sequence of customer contexts $\varphi^{\mathcal{X}} = \{\varphi_i^{\mathcal{X}}(\ell)\}_{\ell\in\mathcal{L}, i\in\mathcal{M}_\ell}$, and satisfies the capacity constraints only in expectation.

Offline LP-based Benchmark-I ($\boldsymbol{LP_1[\varphi^{\mathcal{X}}, w]}$):

$$\begin{aligned} \max_{x} \quad & \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K} r_{ik}(\ell)\, x_{ik}(\ell) \\ \text{s.t.} \quad & \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} s_{ik}(\ell)\, x_{ik}(\ell) \;\le\; \mathcal{C}_k, \quad \forall\, k\in\mathcal{K} \\ & \sum_{k=1}^{K} x_{ik}(\ell) \;\le\; 1, \quad \forall\, \ell\in\mathcal{L},\ \forall\, i\in\mathcal{M}_\ell \\ & x_{ik}(\ell) \;\ge\; 0, \quad \forall\, \ell\in\mathcal{L},\ \forall\, i\in\mathcal{M}_\ell,\ \forall\, k\in\mathcal{K}. \end{aligned}$$

In Lemma II.1 (see Appendix C), we formally prove that the expected total reward of the clairvoyant policy is upper bounded by the above LP-based benchmark.

2.4.2.2 Bayesian Regret of the PRA-LD Algorithm

We state our main theoretical results for the PRA-LD algorithm.

Theorem II.1 (Bayesian Regret of PRA-LD). *The Bayesian regret of the PRA-LD algorithm over the planning horizon of length L is upper bounded with probability at least $1-2\delta$ as:*

$$\textsc{BayesReg}(L) \ \leq H(L,\delta) + c_Q \sigma_\eta K \sqrt{2N_L \log\left(\frac{2}{\delta}\right)} + (1-\alpha)\ \mathbb{E}[V^{BM_1}].$$

In this bound, N_L is the total number of arrivals over L days, and the order of $H(L,\delta)$ is

$$\mathcal{O}\left((d+K)\sqrt{N_L \log\left(1+\frac{N_L}{d+k}\right) \log\left(\frac{1}{\delta^2}+\frac{N_L}{(d+k)\delta^2}\right)} + (d+K)\bar{M}\Delta \log\left(1+\frac{N_L}{d+k}\right) + N_L\delta \right),$$

where $\bar{M}$ is an upper bound on the maximum number of arrivals on each day, and Δ is the maximum number of days required for a feedback to be realized. Also, V^{BM_1} is the total expected reward of the LP-based benchmark-I, and $\mathbb{E}[V^{BM_1}]$ is the expected value of V^{BM_1} over the prior distribution of the unknown model parameter. The term α corresponds to hedging against adversarial arrivals of customer contexts, and it is the constant/expected competitive ratio of the resource allocation mechanism incorporated in the algorithm. That is, defining V^{AUX_1} as the total expected reward obtained by a mechanism for solving the auxiliary problem, we have $V^{AUX_1}/V^{BM_1} \geq \alpha$.

Proof. To obtain the Bayesian Regret, first we define the following notation:

$$V^{BM_1} = \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K} r_{ik}(\ell)\ x^*_{ik}(\ell), \tag{2.9}$$

$$V^{AUX_1} = \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K} r_{ik}(\ell)\ x^{Aux^*}_{ik}(\ell), \tag{2.10}$$

$$V^{ALG_1} = \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K} r_{ik}(\ell)\ x^{Alg^*}_{ik}(\ell), \tag{2.11}$$

where the above three terms respectively denote the total expected reward obtained by (i) the LP-based benchmark-I, which knows the model parameter w as well as the sequence of customer contexts $\varphi^{\mathcal{X}}$ in advance, (ii) the resource allocation mechanism for solving the auxiliary problem, which knows the model parameter w but *does not* know the sequence of customer contexts $\varphi^{\mathcal{X}}$, and (iii) the PRA-LD algorithm (ignoring a penalty term below), which does not know the model parameter w and the sequence of customer contexts $\varphi^{\mathcal{X}}$ in advance.

We should note that the resource consumption is stochastic and the PRA-LD algorithm enforces the capacity constraints only in expectation. Thus, there is a possibility of exceeding the resource capacity in some cases (see Proposition II.3 in Appendix A for details). We impose a penalty on the amount of capacity allocated in excess of the resource capacity and call it PenaltyLoss-I. Thus, the Bayesian regret of PRA-LD can be calculated using the following bridging technique:

$$
\begin{aligned}
\textsc{BayesReg}(L) &= \mathbb{E}\big[V^{BM_1}\big] - \mathbb{E}\big[V^{ALG_1} - \textsc{PenaltyLoss-I}\big] \\
&= \ \mathbb{E}\big[V^{AUX_1} - V^{ALG_1}\big] + \mathbb{E}\big[V^{BM_1} - V^{AUX_1}\big] + \mathbb{E}[\textsc{PenaltyLoss-I}], \qquad (2.12)
\end{aligned}
$$

where the first term is the contextual learning loss associated with learning the distribution of the unknown model parameter related to customers' match quality. The second term is the optimality gap of the resource allocation mechanism used for solving the auxiliary problem. The last term can be viewed as the loss associated with uncertainty in the resource consumption.

In the following, we bound the three terms in (2.12), separately.

Part I (Loss Associated with Stochastic Reward). Based on the definitions of V^{AUX_1} and V^{ALG_1} in (2.10) and (2.11), we have:

$$
\begin{aligned}
\mathbb{E}\big[V^{AUX_1} - V^{ALG_1}\big] &= \mathbb{E}\Big[\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K} r_{ik}(\ell)\, x_{ik}^{Aux^*}(\ell)\Big] - \mathbb{E}\Big[\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K} r_{ik}(\ell)\, x_{ik}^{Alg^*}(\ell)\Big] \\
&= \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K} \mathbb{E}\big[r_{ik}(\ell)\, x_{ik}^{Aux^*}(\ell) - r_{ik}(\ell)\, x_{ik}^{Alg^*}(\ell)\big].
\end{aligned}
$$

The above term is bounded with high probability by the result of Proposition II.2 (see Appendix A). For any $\delta > 0$, the following holds with probability at least $1-\delta$.

$$
\begin{aligned}
&\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K} \mathbb{E}\big[r_{ik}(\ell)\, x_{ik}^{Aux^*}(\ell) - r_{ik}(\ell)\, x_{ik}^{Alg^*}(\ell)\big] \\
&\leq 2\, c_s\sqrt{N_L}\sqrt{2\,(d+K)\log\left(1+\tfrac{c_\phi^2 N_L}{\lambda(d+K)}\right)}\left(\sigma_\xi\sqrt{(d+K)\log\left(1+\tfrac{c_\phi^2 N_L}{\lambda(d+K)}\right)+\log\left(\tfrac{1}{\delta^2}\right)}+\lambda^{1/2}\right) \\
&+ 4\, c_s c_Q (d+K)\bar{M}(1+\Delta)\log\left(1+\frac{c_\phi^2 N_L}{\lambda(d+K)}\right) + c_s c_Q N_L \delta\ .
\end{aligned}
$$

Part II (Loss Associated with Stochastic Resource Consumption). The expected penalty loss can be calculated as follows:

$$\mathbb{E}[\textsc{PenaltyLoss-I}] = c_Q \sum_{k=1}^{K} \mathbb{E}\left[\left(\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \mathcal{S}_{ik}(\ell)\, x_{ik}^{Alg^*}(\ell) - \mathcal{C}_k\right)^{+}\right],$$

where c_Q is the maximum possible value for the match quality.

The above term is bounded with high probability by the result of Proposition II.3 (see Appendix A). For any $\delta > 0$, the following holds with probability at least $1-\delta$.

$$\mathbb{E}[\textsc{PenaltyLoss-I}] \leq c_Q \sigma_\eta K \sqrt{2N_L \log\left(\frac{2}{\delta}\right)}\,.$$

Part III (Loss Associated with Online Resource Allocation). Recall that the auxiliary problem is a simpler version of the actual problem in which the unknown model parameter w is known in advance. To solve the auxiliary problem, one must employ a resource allocation mechanism, for which there is an optimality gap. Suppose α is a constant competitive ratio of the resource allocation mechanism of PRA-LD. Then, given the unknown model parameter w, we have $V^{AUX_1}/V^{BM_1} \geq \alpha$. Next, we have the following by a simple algebra:

$$V^{BM_1} - V^{AUX_1} \leq \big(1-\alpha\big)\, V^{BM_1}.$$

Taking expectation over the prior distribution of w on both sides, we obtain the following:

$$\mathbb{E}\big[V^{BM_1} - V^{AUX_1}\big] \leq (1-\alpha)\, \mathbb{E}[V^{BM_1}]\,.$$

Note that if the competitive ratio α depends on the model parameter, the above result holds by replacing α with its expected value over the prior distribution of the model parameter.

According to (2.12), summing the upper bounds established in Parts I, II, and III completes the proof. □

Corollary II.1 (Bayesian Regret of PRA-LD using Mechanism 1). *With $\delta = \frac{1}{N_L}$, the Bayesian regret of the PRA-LD algorithm with Mechanism 1 over the planning horizon of length L is as follows:*

$$\mathcal{O}\Big((d+K)\sqrt{N_L}\log\big(N_L\big) + (d+K)\bar{M}\Delta\log\big(N_L\big)\Big) + (1-\rho)\,\mathbb{E}[V^{BM_1}],$$

where $\rho = \mathbb{E}[\frac{1-\eta_{\max}\Lambda}{1+\beta}]$, $\eta_{\max} = \max\limits_{i,k,\ell}\{q_{ik}(\ell)\}$, $\Lambda = \max\limits_{i,k,\ell}\big(\frac{s_{ik}(\ell)}{\mathcal{C}_k}\big)$, $\Gamma = (1+\Lambda)^{1/\Lambda}$, and $\beta = \frac{\eta_{\max}}{\Gamma-1}$. When $\Lambda \to 0$ and $\eta_{\max} \to 1$, then coefficient $\beta \to 1/(e-1)$. Thus, the above ratio converges *to $1 - 1/e$, which recovers the classical result in the primal-dual paradigm.*

While the Bayesian regret derived in Theorem II.1 provides a performance guarantee for PRA-LD with a general resource allocation mechanism, this corollary tailors the result to Mechanism 1. The proof hinges on the competitive ratio of Mechanism 1 (see Proposition II.8 in Appendix C) and the result of Theorem II.1 with $\delta = 1/N_L$. The proof is omitted for brevity.

Corollary II.2 (Bayesian Regret of PRA-LD with Unknown Mean Resource Consumption). *With $\delta = \frac{1}{N_L}$, the Bayesian regret of the PRA-LD algorithm when learning both match quality and resource consumption over the planning horizon of length L is as follows:*

$$\mathcal{O}\left((d+K)\sqrt{N_L}\log\left(N_L\right) + (d+K)\bar{M}\Delta\log\left(N_L\right)\right) + (1-\alpha)\,\mathbb{E}[V^{BM_1}].$$

While the Bayesian regret derived in Theorem II.1 provides a performance guarantee for PRA-LD with known mean resource consumption upon arrival of a customer, this corollary tailors the result to a setting in which both match quality and resource consumption should be learned. We assume the following resource consumption model:

If resource $k \in \mathcal{K}$ is chosen for the i^{th} customer on day ℓ, the customer uses $\mathcal{S}_{ik}(\ell)$ units of this resource, regardless of ℓ. We assume that $\mathcal{S}_{ik}(\ell) \in [\underline{c}_S, c_S]$ is a stochastic resource consumption following a linear model with the expected value:

$$\mathbb{E}[\mathcal{S}_{ik}(\ell)] = s_{ik}(\ell) = \langle\, \phi_{ik}(\ell), z\rangle,$$

where $z \in \mathbb{R}^{d+K}$ is the unknown model parameter. The noise values, $\eta_{ik}(\ell) = \mathcal{S}_{ik}(\ell) - \langle\, \phi_{ik}(\ell), z\rangle$, are independent σ_η-sub-Gaussian random variables.

The proof hinges on a new high-probability bound on the penalty loss when the unknown model parameter z should be learned. Following the same steps in Theorem II.1 and with $\delta = 1/N_L$, it can be shown that the loss associated with the stochastic reward (Part I) has the order of $\mathcal{O}\left((d+K)\sqrt{N_L}\log\left(N_L\right) + (d+K)\bar{M}\Delta\log\left(N_L\right)\right)$. When the stochastic resource consumption has a model with an unknown parameter, we prove that the penalty loss (Part II) has the order of $\mathcal{O}\left((d+K)\sqrt{N_L}\log\left(N_L\right) + (d+K)\bar{M}\Delta\log\left(N_L\right)\right)$ (see Proposition II.4 in Appendix A). As expected, this penalty loss is higher compared to the penalty loss when the mean resource consumption is known (i.e., $\mathcal{O}(K\sqrt{N_L\log N_L})$); however, it has the same order as the loss associated with the stochastic reward. Hence, the order of the Bayesian regret remains the same as the one when the mean resource consumption is known. The proof is omitted for brevity.

2.4.2.3 Benchmark for the PAS-LD Algorithm

There are three main differences between our advance scheduling and general resource allocation problems. First, we schedule customers to servers over *multiple* days rather than only assigning customers to resources. Second, some customers may not show up on the service date. Lastly, similar to many service applications, the resources (availability of servers) are perishable. That is, unlike the global resource constraints in the general resource allocation problem, the remaining availability of servers on a day cannot be transferred to the next day. Thus, a different benchmark is needed to evaluate the performance of the PAS-LD algorithm.

Similar to what we argued for our general resource allocation problem, a possible benchmark is the optimal online clairvoyant policy that knows the unknown model parameters and the sequence of customer contexts, but not the realizations of rewards and service times. We employ a stronger offline LP-based benchmark that has access to the same information known by the clairvoyant policy but only needs to satisfy the capacity constraints in expectation. We call it the *offline LP-based benchmark-II* ($LP_2\left[\varphi^{\mathcal{X}}, p, w\right]$), which is a natural LP relaxation of the optimization model (2.5) − (2.8) with $x_{ikt}(\ell) \geq 0$.

Following our arguments in Lemma II.1 (see Appendix C), it can be proven that the expected total reward of the clairvoyant policy is upper bounded by the LP-based benchmark-II. We use this LP model as our benchmark to evaluate the performance of the PAS-LD algorithm.

2.4.2.4 Bayesian Regret of the PAS-LD Algorithm

In Theorem II.2 and Corollary II.3, we state our main theoretical results for the PAS-LD algorithm.

Theorem II.2 (Bayesian Regret of PAS-LD). *The Bayesian regret of the PAS-LD algorithm over the scheduling horizon of length L is upper bounded with probability at least $1 - 3\delta$ as:*

$$\textsc{BayesReg}(L) \leq E(L,\delta) + c_Q \sigma_\eta K L \sqrt{2 N_L \log\left(\frac{2}{\delta}\right)} + (1-\alpha)\,\mathbb{E}[V^{BM_2}].$$

In this bound, N_L is the total number of arrivals over L days, and the order of $E(L,\delta)$ is

$$\mathcal{O}\left((d+A)\sqrt{N_L \log\left(1+\tfrac{N_L}{d+A}\right)\log\left(\tfrac{1}{\delta^2}+\tfrac{N_L}{(d+A)\delta^2}\right)} + (d+A)\bar{M}\Delta \log\left(1+\tfrac{N_L}{d+A}\right) + N_L\delta\right),$$

where $\bar{M}$ is an upper bound on the maximum number of arrivals on each day, and Δ is *the maximum number of days required for a feedback to be realized. Also, V^{BM_2} is the total expected reward of the LP-based benchmark-II, and $\mathbb{E}[V^{BM_2}]$ is the expected value of V^{BM_2} over the prior distribution of the unknown model parameters. The term α is the constant/expected competitive ratio of the advance scheduling mechanism incorporated in the algorithm.*

Proof. To obtain the Bayesian Regret for our PAS-LD algorithm, first we define the following notation:

$$V^{BM_2} = \sum_{\ell=1}^{L} \sum_{i=1}^{M_\ell} \sum_{k=1}^{K} \sum_{t=\ell+1}^{L} r_{ikt}(\ell)\; x^{*}_{ikt}(\ell), \tag{2.13}$$

$$V^{AUX_2} = \sum_{\ell=1}^{L} \sum_{i=1}^{M_\ell} \sum_{k=1}^{K} \sum_{t=\ell+1}^{L} r_{ikt}(\ell)\; x^{Aux^*}_{ikt}(\ell), \tag{2.14}$$

$$V^{ALG_2} = \sum_{\ell=1}^{L} \sum_{i=1}^{M_\ell} \sum_{k=1}^{K} \sum_{t=\ell+1}^{L} r_{ikt}(\ell)\; x^{Alg^*}_{ikt}(\ell), \tag{2.15}$$

where the above three terms respectively denote the total expected reward obtained by (i) the LP-based benchmark-II, which knows the unknown model parameters p and w as well as the sequence of customer contexts $\varphi^{\mathcal{X}}$ in advance, (ii) the advance scheduling mechanism for solving the auxiliary problem, which knows the unknown model parameters but *does not* know the sequence of customer contexts, and (iii) the PAS-LD algorithm (ignoring a penalty term below), which does not know the unknown model parameters and the sequence of customer contexts in advance.

We should note that the service time is stochastic and the PAS-LD algorithm enforces the capacity constraints only in expectation. Thus, there is a possibility of exceeding servers' availability in some cases. We impose a penalty on the amount of capacity allocated in excess of the servers' availability and call it PenaltyLoss-II. Accordingly, the Bayesian regret can be calculated using the following bridging technique:

$$\begin{aligned}\textsc{BayesReg}(L) &= \mathbb{E}\big[V^{BM_2}\big] - \mathbb{E}\big[V^{ALG_2} - \textsc{PenaltyLoss-II}\big] \\ &= \ \mathbb{E}\big[V^{AUX_2} - V^{ALG_2}\big] + \mathbb{E}\big[V^{BM_2} - V^{AUX_2}\big] + \mathbb{E}[\textsc{PenaltyLoss-II}],\end{aligned} \tag{2.16}$$

where the first term is the contextual learning loss associated with learning the distribution of the unknown model parameters, the second term is the optimality gap of the advance scheduling mechanism used for solving the auxiliary problem, and the last term is the loss associated with uncertainty in the service time.

In the following, we bound the three terms in (2.16), separately.

Part I (Loss Associated with Stochastic Reward). Based on the definitions of V^{AUX_2} and V^{ALG_2} in (2.14) and (2.15), we have:

$$\begin{aligned}\mathbb{E}\big[V^{AUX_2} - V^{ALG_2}\big] &= \mathbb{E}\Big[\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\sum_{t=\ell+1}^{L} r_{ikt}(\ell)\; x_{ikt}^{Aux^*}(\ell)\Big] - \mathbb{E}\Big[\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\sum_{t=\ell+1}^{L} r_{ikt}(\ell)\; x_{ikt}^{Alg^*}(\ell)\Big]\\ &= \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\sum_{t=\ell+1}^{L} \mathbb{E}\big[r_{ikt}(\ell)\; x_{ikt}^{Aux^*}(\ell) - r_{ikt}(\ell)\; x_{ikt}^{Alg^*}(\ell)\big].\end{aligned}$$

The above term can be bounded with high probability following the result of Proposition II.6 (see Appendix B). For any $\delta > 0$, the following holds with probability at least $1 - 2\delta$.

$$\begin{aligned}&\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\sum_{t=\ell+1}^{L} \mathbb{E}\big[r_{ikt}(\ell)\; x_{ikt}^{Aux^*}(\ell) - r_{ikt}(\ell)\; x_{ikt}^{Alg^*}(\ell)\big]\\ &\leq 2\; c_s\sqrt{N_L}\sqrt{2\;(d+A)\log\left(1+\tfrac{c_\phi^2 N_L}{\lambda(d+A)}\right)}\left((c_Q+\sigma_\xi)\sqrt{(d+A)\log\left(1+\tfrac{c_\phi^2 N_L}{\lambda(d+A)}\right)+\log\left(\tfrac{1}{\delta^2}\right)}+\lambda^{1/2}\right)\\ &+ 4\; c_s c_Q (d+A)\bar{M}(1+\Delta)\log\left(1+\frac{c_\phi^2 N_L}{\lambda(d+A)}\right) + 2\; c_s c_Q N_L \delta.\end{aligned}$$

Part II (Loss Associated with Stochastic Service Time). We assume that if a customer does not show up on the service date, the server's availability is still decreased by the expected service time of the customer. Accordingly, the expected penalty loss can be calculated as follows:

$$\begin{aligned}&\mathbb{E}[\textsc{PenaltyLoss-II}] =\\ &c_Q\sum_{k=1}^{K}\sum_{t=2}^{L}\mathbb{E}\left[\left(\sum_{\ell=1}^{t-1}\sum_{i=1}^{M_\ell}\Big(\mathcal{SU}_i(\ell)\;\mathcal{S}_{ikt}(\ell) + \big(1-\mathcal{SU}_i(\ell)\big)s_{ikt}(\ell)\Big)\; x_{ikt}^{Alg^*}(\ell) - \mathcal{C}_{kt}\right)^+\right].\end{aligned}$$

The above term can be bounded with high probability following the result of Proposition II.7 (see Appendix B). For any $\delta > 0$, the following holds with probability at least $1 - \delta$.

$$\mathbb{E}[\textsc{PenaltyLoss-II}] \leq c_Q\sigma_\eta KL\sqrt{2N_L\log\left(\frac{2}{\delta}\right)}\;.$$

Part III (Loss Associated with Online Advance Scheduling). The auxiliary problem is a simpler version of the actual problem in which the unknown model parameters p and w are known in advance. To solve the auxiliary problem, there is a need for an advance scheduling mechanism. Similar to Part III of Theorem II.1, the optimality gap of this mechanism for solving the auxiliary problem can be bounded using the concept of competitive ratio as follows:

$$\mathbb{E}\big[V^{BM_2} - V^{AUX_2}\big] \leq (1-\alpha)\,\mathbb{E}[V^{BM_2}],$$

where the expectation is over the prior distribution of the unknown model parameters, and α is the constant/expected competitive ratio of the advance scheduling mechanism of PAS-LD.

According to (2.16), summing the upper bounds established in Parts I, II, and III completes the proof. □

Corollary II.3 (Bayesian Regret of PAS-LD with Mechanism 2). *With $\delta = \frac{1}{N_L}$, the Bayesian regret of the PAS-LD algorithm with Mechanism 2 over the scheduling horizon of length L is as follows:*

$$\mathcal{O}\left((d+A)\sqrt{N_L}\log\big(N_L\big) + (d+A)\bar{M}\Delta\log\big(N_L\big) + KL\sqrt{N_L\log\big(N_L\big)}\right) + (1-\widetilde{\rho})\,\mathbb{E}[V^{BM_2}],$$

where $\widetilde{\rho} = \mathbb{E}[\frac{1-\widetilde{\eta}_{\max}\widetilde{\Lambda}}{1+\widetilde{\beta}}]$, $\widetilde{\eta}_{\max} = \max\limits_{i,k,t,\ell}\{q_{ikt}(\ell)\}$, $\widetilde{\Lambda} = \max\limits_{i,k,t,\ell}\big(\frac{s_{ikt}(\ell)}{\mathcal{C}_{kt}}\big)$, $\widetilde{\Gamma} = (1+\widetilde{\Lambda})^{1/\widetilde{\Lambda}}$, and $\widetilde{\beta} = \frac{\widetilde{\eta}_{\max}}{\widetilde{\Gamma}-1}$. When $\widetilde{\Lambda} \to 0$ and $\widetilde{\eta}_{\max} \to 1$, then coefficient $\widetilde{\beta} \to 1/(e-1)$. Thus, the above ratio converges to $1 - 1/e$.

The proof hinges on the competitive ratio of Mechanism 2 (see Proposition II.9 in Appendix C) and the result of Theorem II.2 with $\delta = 1/N_L$. The proof for the competitive ratio of Mechanism 2 can be reproduced by working with dual problem of $LP_2\,[\varphi^{\mathcal{X}}, p, w]$ instead of $LP_1\,[\varphi^{\mathcal{X}}, w]$ in Proposition II.8 (see Appendix C). The proof is omitted for brevity.

2.4.3 Discussions of the Main Results

We discuss our theoretical results and position them in the literature.

First, as a by-product of our regret analysis, we develop a new confidence bound for the unknown parameter in a linear model under bounded delayed feedback. The effect of delayed feedback in contextual bandits is much less explored in the literature than the other settings, but we note the studies of [45] (with fixed delay) and [112] (with bounded delay). The latter studied a GLM bandit with delay and proposed the delayed UCB algorithm with a warm-up period and a non-regularized MLE. When there is only an upper bound on delay, they established a regret bound of $\tilde{\mathcal{O}}\big((d+\sqrt{dD_{\max}})\sqrt{T}\big)$, where $D_{\max}$ is an upper bound on delays and d is the feature dimension. For the contextual bandit problem studied by [112] when the reward model is linear and the feedback delay is bounded, our confidence bound constructed using an OLS estimator with regularization yields a regret bound of $\tilde{\mathcal{O}}\big(d\sqrt{T} + dD_{\max}\big)$. Similar to the regret bound of [112], the first term of our regret bound does not depend on delay and the upper bound on delays only impacts the second term. Note that their regret bound $\tilde{\mathcal{O}}\big((d + \sqrt{dD_{\max}})\sqrt{T}\big)$ is obtained under the implicit assumption that $dD_{\max} \leq T$. Under this assumption, our regret bound is strictly tighter than the one derived by [112].

In our advance scheduling problem, customers may not show up for the scheduled service after being assigned to a server and a service date. This adds on an additional layer of complexity in learning the match quality because the match quality feedback of a customer cannot be observed at all if the customer does not show up on the service date. Our PAS-LD algorithm learns the distribution of pw vector, where p and w are the unknown model parameters corresponding to no-show and match quality. We developed an estimator for pw and provided a new confidence bound for pw under delayed feedback. Compared to the confidence bound derived for w, this confidence bound has an extra additive logarithmic term in the number of arrivals, which can be viewed as the cost of estimating pw instead of w.

Next, we shall relate to the literature of contextual MAB with resource constraints. In this area, the studies of [12], [3], and [5] are among the closely related ones to our work. However, their online algorithms cannot be extended to our setting because of their IID assumption for context vectors of customers that results in the existence of an underlying (fixed) optimal randomized allocation strategy. [35] considered the allocation of limited resources to heterogeneous customers over time without the IID assumption. In their setting, the rewards are known and the resource consumption distribution associated with each customer type and action should be learned over time. They provided an information-theoretic lower bound that depends on the competitive ratio of an oracle for solving a problem similar to our auxiliary problem. Their methodology and techniques are different and not comparable to our work. Our online algorithms admit Bayesian regret upper bounds without needing the IID assumption. Our Bayesian regret bounds indicate that the contextual learning loss associated with learning the distributions of the unknown model parameters is sub-linear in the number of arrivals over the planning horizon. The loss associated with the uncertainty in the resource consumption values is also sub-linear in the number of arrivals over the planning horizon. The sequence of customer contexts is adversarial in our setting and it comes at the cost of having the term $(1-\alpha)\,\mathbb{E}[V^{BM}]$ in the bound. This term is indeed the optimality gap of a resource allocation mechanism with a competitive ratio of α for solving the auxiliary problem.

In the PRA-LD algorithm, the loss associated with contextual learning with delay is of order $\mathcal{O}\left((d+K)\sqrt{N_L}\log(N_L) + (d+K)\bar{M}\Delta\log(N_L)\right)$, which can be further decomposed into two terms. The first term $\mathcal{O}\left((d+K)\sqrt{N_L}\log(N_L)\right)$ has the same order as the state-of-the-art regret bound for the contextual learning problems, and the second term $\mathcal{O}\left((d+K)\bar{M}\Delta\log(N_L)\right)$

comes from the assumption of bounded delayed feedback. The loss associated with uncertainty in the resource consumption values is of order $\mathcal{O}\left(K\sqrt{N_L \log(N_L)}\right)$. Also, the loss associated with the optimality gap of Mechanism 1 is $\left(1 - \mathbb{E}\left[\frac{1-\eta_{max}\Lambda}{1+\beta}\right]\right)\ \mathbb{E}[V^{BM_1}]$, where $\eta_{\max} = \max\limits_{i,k,\ell}\{q_{ik}(\ell)\}$, $\Lambda = \max\limits_{i,k,\ell}\left(\frac{s_{ik}(\ell)}{\mathcal{C}_k}\right)$, $\Gamma = (1+\Lambda)^{1/\Lambda}$, and $\beta = \frac{\eta_{\max}}{\Gamma-1}$. From a different angle, $\frac{1-\eta_{max}\Lambda}{1+\beta}$ can be viewed as the competitive ratio of Mechanism 1 for online resource allocation. When $\eta_{\max}$ tends to 1 and the ratio of expected resource consumption per request to total capacity Λ tends to 0, the parameter $\beta = \frac{\eta_{\max}}{\Gamma-1}$ tends to $1/(e-1)$ because $\Gamma = (1+\Lambda)^{1/\Lambda}$; subsequently, the competitive ratio converges to $1 - 1/e$ which recovers the classical result obtained by [81] and [25]. The same arguments and interpretations also hold for the Bayesian regret bound derived for PAS-LD.

2.5 Case Study and Empirical Results

Our models and methodology are motivated by emerging technology and intent in the healthcare industry to offer online advance scheduling to patients. Online learning facilitates the successful adaption of new approaches and supports learning patient preferences. Our methodology can increase access to timely and appropriate high-quality care by offering personalized appointment visits based on patients' needs and preferences. Using clinical data from a partner health system, we evaluate the performance of our proposed PAS-LD algorithm compared with other algorithms/policies.

2.5.1 Data Description and Problem Formulation

Data Description. Our partner health system offers appointment visits to provide diagnosis, consultation, and several procedures, including prostate cancer, micro-surgical urology, kidney stones/cancer, and bladder cancer. The medical clinic is staffed with physicians (MDs) and physician assistants (PAs) with five major specialties: (i) general, (ii) andrology (Andro), (iii) oncology (Onco), (iv) endoscopy (Endo), and (v) neurourology and pelvic reconstructive surgery (NPR). We used the patient appointments related to 12 providers (i.e., 8 MDs and 4 PAs) working in the same medical clinic five days a week from 8:00 am to 5:00 pm.

Our clinical dataset contains more than 4500 appointment visits for which we have information regarding both patients and providers. Each visit provides: (1) age, (2) chief complaint of the patient (service request), (3) patient's urgency level (Emergent, Urgent, Elective), (4) arrival date and scheduled date providing the appointment delay, (5) service time/duration, (6) provider for the patient, and (7) provider's credentials (MD/PA). We

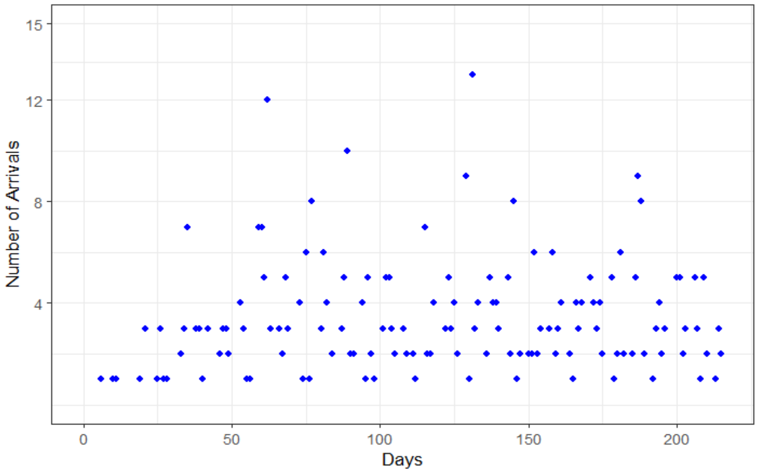


Figure 2.1: Total number of patient arrivals with NPR service request, which is served by two providers.

selected several important variables which can be classified into the following two major categories:

- *Patient characteristics*: Demographics, service type request (General, Andro, Onco, Endo, and NPR), and urgency level (Emergent, Urgent, Elective),
- *Provider skills*: Credentials (MD/PA) and expertise level for the relevant specialty.

As there is no system for scoring the expertise of providers in the medical clinic at the time, we employed judgment to create one. Consulting with the clinic and considering the main- and sub-specialties of each provider, we assigned a value between zero and one to the relevant five major specialties for each provider. In some settings, this information could be computed using providers' ratings by patients. The no-show events were modeled as independent Bernoulli trials.

As we mentioned, the medical clinic offers appointment visits to patients with five major service requests. Our data shows that there is high variability in the arrival pattern of patients over different days. As an example, the number of patients who arrive on each day with an NPR service request is shown in Figure 2.1. As can be seen, the arrival pattern is highly uncertain and the same uncertainty also applies to the arrival pattern of patients with other service requests. Thus, a data-driven online algorithm that does not require information regarding the distribution of arrivals or patterns over time is intuitively preferable to traditional approaches.

Problem Formulation. We formulate this patient appointment scheduling problem using our framework for advance scheduling. We make no assumption on future arrivals. Upon arrival of a patient, either a provider and an appointment date should be assigned to the patient or the patient must be rejected. Patient rewards depend on the patient-provider match quality, service time, and the no-show outcome. The quality of match could be obtained by patient satisfaction surveys to better address the patients' needs. It is worth noting that the priority categories of emergent, urgent, and elective patients (urgency level) paired with appointment delay provide the opportunity to indirectly reduce access delay for the more urgent patients.

The main considerations for assigning a provider-date to a patient are the available capacity of the clinic, the expected patient-specific service time, and the expected patient-specific match quality. To calculate the expected service time, we need the information on patient characteristics and the skills of the providers. The expected match quality also needs a third element which is the appointment delay (i.e., number of days between the arrival date and the appointment date). The feedback information on the match quality is assumed to be revealed immediately after the patient is served by the provider on the appointment date. The no-show feedback is revealed immediately at the time of the scheduled appointment. The expected patient-specific service times are estimated by a linear regression model trained on the entire dataset. The objective of the algorithm is to maximize the total expected reward over a finite-time scheduling horizon.

Real-time evaluation is the ideal way to assess our online algorithm. Evaluating our algorithm with offline clinical data has two main hurdles. First, our algorithm has not been implemented by our partner health system; thus, there is a lack of data on the patient-provider match quality and we must estimate it. We randomly generated the patient-provider match quality outcomes using different uniform distributions as a function of the patient characteristics, the skills of the chosen provider, and the appointment delay. Second, even if we had access to the real patient-provider match quality, evaluating our algorithm retrospectively based on the observational data would be challenging since we need counterfactual outcomes. In practice, the outcomes of the other decisions not taken are not required, because the algorithm assigns a patient to only one provider-date pair and obtains the corresponding outcome. However, we need to estimate the counterfactuals when we evaluate our algorithm based on the observational data. In particular, if a patient is assigned to a provider-date in the dataset but our algorithm assigns the patient to a different provider-date, we must estimate the outcome associated with our algorithm's decision to evaluate its performance. We separately estimated the counterfactuals corresponding to patient-provider match quality by a linear regression model trained on the entire dataset.

2.5.2 Evaluation and Empirical Results

Using our clinical data, we evaluate the performance of the PAS-LD algorithm with respect to the commonly used First-Come-First-Served (FCFS) policy and other algorithms. We consider 10 random permutations of our dataset to ensure the result is not tied to one realization of patient arrivals. We calculate two performance measures: average performance and average cumulative regret. The *average performance* is the ratio of the cumulative expected reward obtained by an algorithm/policy relative to the optimal objective function value of the LP-based benchmark-II averaged over permutations. The *average cumulative regret* measures the difference between the cumulative expected reward of the offline LP-based benchmark-II and an algorithm/policy averaged over permutations.

Since providers' availability hours are limited in our clinic, we cannot easily compare our algorithm with the most widely used online learning algorithms in the literature. We design and compare the following algorithms/policies:

(a) *PAS-LD algorithm*: The online algorithm generated by our framework with Mechanism 2 for online advance scheduling.

(b) *Greedy algorithm*: The PAS-LD algorithm with the greedy advance scheduling mechanism in which a patient should be assigned to a provider-date that yields the highest reward.

(c) *FCFS policy*: A pervasive policy implemented for many service systems. This policy assigns each patient to a provider with the earliest availability as long as it meets the availability of the provider. Ties are broken randomly.

(d) *PAS-LD-OFU algorithm*: A variant of the PAS-LD algorithm with Mechanism 2 to replace the PS method with the optimisim in the face of uncertainty (OFU) method (see [76]).

Average Performance. In this analysis, we aim to gain insights into the sensitivity of the average performance with respect to variations in the capacity (providers' availability hours) and the mean service time. We consider a scheduling horizon of 50 days and compare the average performance of PAS-LD, Greedy, and FCFS under different capacity and mean service time scales.

Table 2.1 reports the average performance results with regard to different levels of a capacity scale. In particular, we evaluate the average performance with regard to a capacity scale parameter $c_1 \in \{1.5, 1.4, \ldots, 0.7\}$, where c_1 is the multiplier on the hours available

Capacity scale	**FCFS**	**Greedy**	**PAS-LD**
1.5	73.95%	89.68%	90.83%
1.4	74.13%	89.03%	90.57%
1.3	73.23%	85.72%	90.31%
1.2	72.56%	85.36%	89.69%
1.1	71.51%	83.28%	89.36%
1.0	69.68%	80.61%	88.77%
0.9	64.96%	77.05%	88.95%
0.8	61.39%	72.97%	87.19%
0.7	57.68%	69.18%	87.59%

Table 2.1: Average performance of algorithms/policies as capacity is varied.

for every provider. The results indicate that when the capacity increases, the average performance increases. As the capacity decreases, the average performance decreases. This insight is intuitive and consistent with our theoretical result. Predictably, PAS-LD has the best average performance among the other policies/algorithms because (i) it leverages the estimated patient-specific match quality, and (ii) it uses an advance scheduling mechanism that hedges against the future arrival sequence by judiciously allocating the scarce resources over the scheduling horizon. Both FCFS and Greedy are more sensitive to variations in capacity compared to PAS-LD (see Table 2.3). The poor performance of FCFS is largely due to the lack of attention to the reward of arriving patients during the decision-making process. FCFS has better performance when providers' availability hours are increased since there is less need to judiciously allocate the scarce resources in that case. Greedy performs consistently better than FCFS and also performs comparably to PAS-LD when the capacity is abundant. Note that both PAS-LD and Greedy learn the expected patient match quality adaptively and take it into account for scheduling patient appointments. Assigning a patient to a provider-date that yields the highest reward (being greedy) can provide a good strategy when there is abundant capacity because hedging against the future arrival sequence becomes less important. The overall results show that the PAS-LD algorithm offers the greatest value when the capacity is moderate to low.

Table 2.2 reports the average performance results with regard to different levels of a mean service time scale. In particular, we evaluate the average performance with regard to a mean service time scale parameter $c_2 \in \{0.7, 0.8, \ldots, 1.5\}$, where c_2 is the multiplier on the mean service time for every patient. The results are consistent with the results reported in Table 2.1 in a sense that PAS-LD outperforms FCFS and Greedy over all values of the mean service time scale parameter. We observe that the average performance of PAS-LD is less

Service time scale	**FCFS**	**Greedy**	**PAS-LD**
0.7	74.05%	88.33%	90.67%
0.8	72.56%	84.32%	89.38%
0.9	71.51%	83.19%	89.30%
1.0	69.68%	80.61%	88.77%
1.1	65.07%	80.01%	88.51%
1.2	62.00%	78.92%	88.50%
1.3	61.25%	74.95%	88.45%
1.4	57.46%	73.77%	88.21%
1.5	57.39%	70.57%	88.10%

Table 2.2: Average performance of algorithms/policies as mean service time is varied.

Measure	**Variation in capacity**			**Variation in service time**		
	FCFS	Greedy	PAS-LD	FCFS	Greedy	PAS-LD
Mean	68.79%	81.43%	89.25%	65.66%	79.41%	88.88%
Standard deviation	6.02%	7.11%	1.27%	6.49%	5.59%	0.81%
Range	16.45%	20.50%	3.64%	16.66%	17.76%	2.57%

Table 2.3: Statistical measures corresponding to variations in capacity and mean service time.

sensitive to variations in the mean service time compared to FCFS and Greedy (see Table 2.3). Note that when the mean service time is short, the average performance increases. As the mean service time increases, the average performance decreases. This insight is not only consistent with our theoretical result but also intuitive. Increasing the mean service time, PAS-LD will have less flexibility in assignment decisions which results in lower average performance. Note that for the next two analyses we return to the real-world capacity and mean service time (scales of $c_1 = 1$ and $c_2 = 1$).

Average Cumulative Regret. In this analysis, we first evaluate the performance of three strategies to deal with delayed feedback. Next, we investigate another method to balance the exploration-exploitation trade-off and evaluate its impact on the performance of the PAS-LD algorithm.

We consider the following three strategies to deal with delayed feedback and discuss the corresponding results. The first one is our proposed *asynchronous* (ASYN) strategy in which we update the estimator for the unknown model parameters on the fly. The second and third ones are *fixed waiting time* (FWT) strategies in which we update the estimator

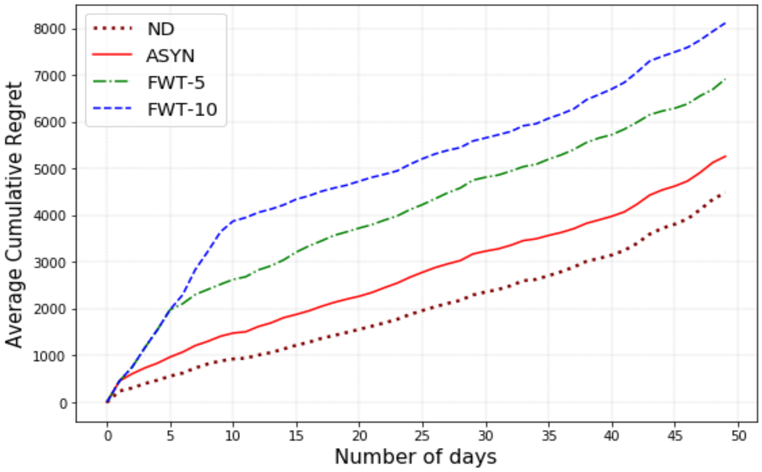


Figure 2.2: Average cumulative regret of different updating strategies.

based on the feedback of a decision after a fixed number of days (c_{FWT}) that is long enough to guarantee that the feedback is realized. That is, we update the estimator on day ℓ by supplying the feedback of decisions made on day $\ell - c_{FWT}$ and assume that the feedback outcomes are realized by this time. We consider two FWT strategies, including (i) FWT-5: the maximum time for each feedback to get realized is 5 days, and (ii) FWT-10: similar to the previous scenario but for 10 days. We also consider the ideal case in which there is no delayed feedback (ND). That is, we assume that a patient's feedback is realized *immediately* after assigning the patient to a provider-date.

Figure 2.2 illustrates the average cumulative regret of the variants of PAS-LD with different strategies to deal with delayed feedback as well as the ideal case with no delay. The ideal case, ND, outperforms the others since this is an unrealistic case with immediate feedback at the time of making the appointment. As can be seen, the ASYN strategy implemented in our actual PAS-LD algorithm is not much worse than ND and outperforms the two FWT strategies. When there are few samples, the slope of the average cumulative regret of the ASYN strategy is significantly less than the others because it learns more quickly using the realized feedback outcomes rather than the FWT strategies which receive the same information but with delay. As we get more samples and the estimator gets closer to convergence, the slopes of ASYN, FWT-5, and FWT-10 converge to the slope of the ideal case.

In our PAS-LD algorithm, the exploration-exploitation trade-off is balanced by using the PS method. OFU is another popular method used to balance the exploration-exploitation trade-off in online learning algorithms. The PAS-LD-OFU algorithm with the OFU method constructs confidence sets for the unknown model parameters and selects the action with the highest optimistic estimate (highest upper bound). We evaluate the impact of using PS and

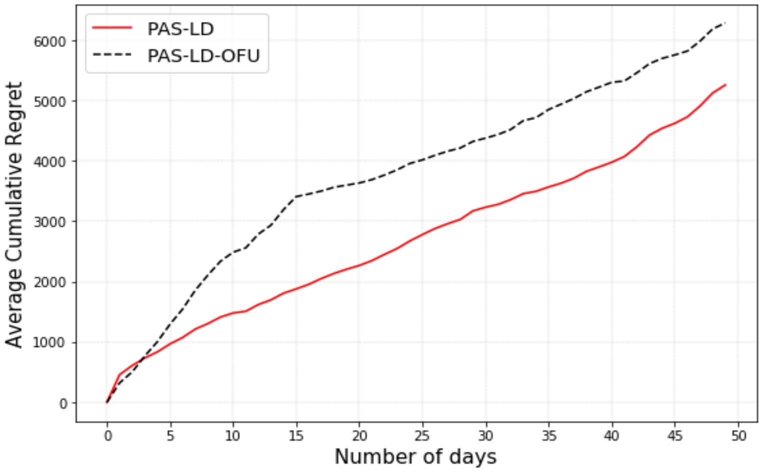


Figure 2.3: Average cumulative regret of PAS-LD and PAS-LD-OFU.

OFU methods by comparing the average cumulative regrets obtained by PAS-LD-OFU and PAS-LD. To make a fair comparison, we consider a non-informative prior for the unknown model parameters.

Figure 2.3 shows the average cumulative regret of PAS-LD and PAS-LD-OFU over 50 days. Overall, the average cumulative regret of PAS-LD is less than that of PAS-LD-OFU. When there are few samples, the slope of the average cumulative regret of PAS-LD-OFU is significantly higher than PAS-LD, but as we get more samples the slope of PAS-LD-OFU gets closer to the slope of PAS-LD. This implies that the PS method used in PAS-LD yields a lower regret in our problem compared to the OFU method used in PAS-LD-OFU, especially when there are few samples. The better learning ability of algorithms with the PS method compared to algorithms with the OFU method has been seen empirically in other problems as well (see, e.g., [29]). We further investigate the difference in the performance of PAS-LD and PAS-LD-OFU by looking at their average cumulative regrets and the corresponding ratios with respect to different strategies to deal with delayed feedback. Table 2.4 shows the average cumulative regret of PAS-LD-OFU and PAS-LD after 50 days corresponding to the ASYN strategy and three FWT strategies (i.e., FWT-5, FWT-10, and FWT-15). We find that PAS-LD is more robust compared to PAS-LD-OFU as the ratio between the average cumulative regrets is increasing when the fixed waiting time is getting longer. The advantage of PS over OFU shown in this analysis might be due to the randomization process used in the PS method and its ability to better alleviates the influence of delayed feedback outcomes compared to the OFU method with optimistic estimates.

Algorithm	**Strategy**			
	ASYN	FWT-5	FWT-10	FWT-15
PAS-LD-OFU	6286.63	8849.46	10709.50	12672.51
PAS-LD	5318.34	6990.46	8203.40	9285.01
Ratio	1.18	1.27	1.31	1.36

Table 2.4: Impact of different learning methods on average cumulative regret at the end of the horizon.

2.6 Conclusion

We studied a resource allocation problem with personalized learning, where the sequence of customer contexts is adversarial and the customer reward and the resource consumption are stochastic and unknown. Our objective was to create algorithmic optimization procedures with theoretical performance guarantees, especially to allow healthcare delivery systems make a real-time decision to allocate each arriving patient to a suitable clinician in a timely manner. Learning is emphasized here to solve settings with a lack of historical data. We introduced a generic framework for adversarial arrivals, which judiciously synergizes online learning with a broad class of online resource allocation mechanisms. Within this framework, we developed two online algorithms, namely PRA-LD and PAS-LD, which admit rigorous performance guarantees. While in the general resource allocation model, customers should be assigned to only resources, they should be assigned to servers and service dates in the advance scheduling model. The PRA-LD algorithm is designed for a general resource allocation problem in which customers arrive sequentially and should be either assigned to a resource or rejected. The PAS-LD algorithm is designed for a multi-day advance scheduling problem and it offers servers and service dates to sequentially arriving customers while accounting for the possibility of no-show. Our algorithms strike a three-way balance between exploration, exploitation, and hedging against the future arrival sequence. They can operate under practical settings for which the learning process is conducted under delayed feedback outcomes.

We provided theoretical performance guarantees for the proposed algorithms, which require several new technical ideas some of which may be of independent interest beyond the scope of this study. We also evaluated the empirical performance of our PAS-LD algorithm with a primal-dual advance scheduling mechanism using a dataset from our partner health system. The empirical results showed that the proposed PAS-LD algorithm with a primal-dual advance scheduling mechanism performs better than a greedy variant of it and extremely well compared to the pervasive FCFS policy. We also found that our advance scheduling algorithm can handle no-shows and delayed feedback very well, which are among the major practical challenges in healthcare settings. Our asynchronous strategy for updating the estimators of unknown model parameters outperformed other strategies by a large margin and performed not much worse than the ideal case without delayed feedback. Our framework and proposed algorithms can be applied to many other practical problems that require both learning and resource allocation in an online fashion.

2.7 Appendix

2.7.1 Appendix A. Technical Results for the PRA-LD Algorithm

Proposition II.1 (Confidence Bound for PRA-LD under Delayed Feedback). *For any i, ℓ, and $\delta > 0$, the following holds with probability at least $1-\delta$.*

$$\begin{aligned}&\left|\langle\, \phi_{ik^*}(\ell), w\rangle - \langle\, \phi_{ik^*}(\ell), \hat{w}_i(\ell)\rangle\right| \\ &\leq \|\phi_{ik^*}(\ell)\|_{V_{i\ell}^{-1}} \left(\sigma_\xi \sqrt{2\log\left(\frac{\det(V_{i\ell})^{1/2}\det(\lambda I)^{-1/2}}{\delta}\right)} + \Sigma_{ik^*}^{(w)}(\ell)\right),\end{aligned}$$

where $V_{i\ell} = \begin{cases} \sum_{s=1}^{\ell-1}\sum_{j=1}^{M_s} \phi_{jk^*}(s)\ \phi'_{jk^*}(s) + \lambda I & i=1 \\ \sum_{s=1}^{\ell-1}\sum_{j=1}^{M_s} \phi_{jk^*}(s)\ \phi'_{jk^*}(s) + \sum_{j=1}^{i-1} \phi_{jk^*}(\ell)\ \phi'_{jk^*}(\ell) + \lambda I & i \geq 2 \end{cases}$ *is the design matrix such that* $\lambda > 0$ *and*
$\Sigma_{ik^*}^{(w)}(\ell) = c_Q\Big(\sum_{s=\max\{1,\ell-\Delta\}}^{\ell-1}\sum_{j=1}^{M_s} \|\phi_{jk^*}(s)\|_{V_{i\ell}^{-1}} + \big(\sum_{j=1}^{i-1}\|\phi_{jk^*}(\ell)\|_{V_{i\ell}^{-1}}\big)\mathbb{1}_{i\neq 1}\Big) + \lambda^{1/2}.$

Proof. Constructing a confidence bound on the expected match quality involves the following two steps.

Step 1 (Online estimator for parameter w). Recall that we assumed $\mathcal{Q}_{ik^*}(\ell)$ as the uncertain feedback of the i^{th} customer on day ℓ assigned to resource k^*. Based on the least squares principle, we can obtain an estimator for w by minimizing the following term:

$$\mathcal{U}_{i\ell}(w) = \sum_{s=1}^{\ell-1}\sum_{j=1}^{M_s}\left(\phi'_{jk^*}(s)\ w - \mathcal{Q}_{jk^*}(s)\right)^2 + \Big(\sum_{j=1}^{i-1}\left(\phi'_{jk^*}(\ell)\ w - \mathcal{Q}_{jk^*}(\ell)\right)^2\Big)\mathbb{1}_{i\neq 1} + \lambda\left\|w\right\|^2,$$

where $\lambda > 0$ is the regularization parameter. Note that for the first customer of each day ($i=1$), $\mathcal{U}_{i\ell}(w)$ includes all the feature vectors and the realized feedback outcomes up to the beginning of day ℓ. However, for other customers ($i \neq 1$), $\mathcal{U}_{i\ell}(w)$ includes all the feature vectors and the realized feedback outcomes up to the beginning of day ℓ plus all the feature vectors and the feedback outcomes observed on day ℓ until the $i-1^{th}$ customer.

Minimizing the above term ($\nabla_w \mathcal{U}_{i\ell}(w) = 0$) yields the following estimator for the parameter w:

$$\hat{w}_i^I(\ell) = \left(\sum_{s=1}^{\ell-1} \sum_{j=1}^{M_s} \phi_{jk^*}(s)\ \phi_{jk^*}^{'}(s) + \big(\sum_{j=1}^{i-1} \phi_{jk^*}(\ell)\ \phi_{jk^*}^{'}(\ell) \big) \mathbb{1}_{i\neq 1} + \lambda I \right)^{-1}$$
$$\left(\sum_{s=1}^{\ell-1} \sum_{j=1}^{M_s} \mathcal{Q}_{jk^*}(s)\ \phi_{jk^*}(s) + \big(\sum_{j=1}^{i-1} \mathcal{Q}_{jk^*}(\ell)\ \phi_{jk^*}(\ell) \big) \mathbb{1}_{i\neq 1} \right),$$

where $\hat{w}_i^I(\ell)$ is the *ideal* L^2-regularized least squares estimator of w with the regularization parameter λ. This estimator is obtained by *assuming* that all the match quality feedback outcomes of prior customers are realized by the time we calculate $\hat{w}_i^I(\ell)$.

In our setting, we have access to the feature vectors of all customers assigned to different resources (i.e., feature vectors of customers will be available right after making decisions for them) but the feedback outcomes might not be available immediately after making decisions for them. Indeed, feedback outcomes arrive sequentially with delay in our setting. Accordingly, it is not possible to use $\hat{w}_i^I(\ell)$ as our estimator because it requires the assumption of no delayed feedback. In the PRA-LD algorithm, we use our asynchronous strategy in which estimators for the unknown model parameters are updated on the fly based on the *available* information. Let (ℓ, i, k^*) be a tuple referring to the i^{th} customer on day ℓ assigned to resource k^*. We define $\mathcal{RF}^{(\mathcal{Q})}(i,\ell)$ as the set containing tuples (s, j, k^*) of customers with *realized* match quality feedback outcomes before the arrival of the i^{th} customer on day ℓ, where $s \leq \ell$ and $j < i$. Similarly, we define $\mathcal{UF}^{(\mathcal{Q})}(i,\ell)$ as the set containing tuples (s, j, k^*) of customers with *unrealized* match quality feedback outcomes before the arrival of the i^{th} customer on day ℓ.

We define $\hat{w}_i(\ell)$ as our new estimator, which uses only the *available* information up to the current time. It can be obtained as follows:

$$\hat{w}_i(\ell) = \left(\sum_{s=1}^{\ell-1} \sum_{j=1}^{M_s} \phi_{jk^*}(s)\ \phi_{jk^*}^{'}(s) + \big(\sum_{j=1}^{i-1} \phi_{jk^*}(\ell)\ \phi_{jk^*}^{'}(\ell) \big) \mathbb{1}_{i\neq 1} + \lambda I \right)^{-1}$$
$$\left(\sum_{(s,j,k^*) \in \mathcal{RF}^{(\mathcal{Q})}(i,\ell)} \mathcal{Q}_{jk^*}(s)\ \phi_{jk^*}(s) \right).$$

Note that we update this estimator after each customer by using the available information up to the current time. In particular, the estimator always uses the feature vectors of all customers assigned, but it uses the feedback outcomes if they are realized up to the current time.

Step 2 (Confidence bound). Our aim is to construct a confidence bound that contains the true expected match quality with high probability. To do so, we first establish the following decomposition:

$$w - \hat{w}_i(\ell) = \big(w - \hat{w}_i^I(\ell)\big) + \big(\hat{w}_i^I(\ell) - \hat{w}_i(\ell)\big)\,.$$

We would like to derive an upper bound for $\big|\langle\, \phi_{ik^*}(\ell), w\rangle - \langle\, \phi_{ik^*}(\ell), \hat{w}_i(\ell)\rangle\big|$. According to the above decomposition and using the triangle inequality, we have:

$$\begin{aligned}&\big|\langle\, \phi_{ik^*}(\ell), w\rangle - \langle\, \phi_{ik^*}(\ell), \hat{w}_i(\ell)\rangle\big| \leq \\ &\underbrace{\big|\langle\, \phi_{ik^*}(\ell), w\rangle - \langle\, \phi_{ik^*}(\ell), \hat{w}_i^I(\ell)\rangle\big|}_{\text{(I)}} + \underbrace{\big|\langle\, \phi_{ik^*}(\ell), \hat{w}_i^I(\ell)\rangle - \langle\, \phi_{ik^*}(\ell), \hat{w}_i(\ell)\rangle\big|}_{\text{(II)}}\,.\end{aligned}$$

Next, we construct the confidence bound by bounding the two terms.

Term (I): Following the method used in [1] (see Theorem 2), it can be shown that the following bound holds for any i and ℓ:

$$\begin{aligned}&\big|\langle\, \phi_{ik^*}(\ell), w\rangle - \langle\, \phi_{ik^*}(\ell), \hat{w}_i^I(\ell)\rangle\big| \\ &\leq \|\phi_{ik^*}(\ell)\|_{V_{i\ell}^{-1}} \left(\left\| \sum_{s=1}^{\ell-1}\sum_{j=1}^{M_s} \xi_{jk^*}(s)\ \phi_{jk^*}(s) + \big(\sum_{j=1}^{i-1} \xi_{jk^*}(\ell)\ \phi_{jk^*}(\ell)\big)\mathbb{1}_{i\neq 1} \right\|_{V_{i\ell}^{-1}} + \lambda^{1/2} \right),\end{aligned}$$

where $V_{i\ell} = \begin{cases} \sum_{s=1}^{\ell-1}\sum_{j=1}^{M_s} \phi_{jk^*}(s)\ \phi'_{jk^*}(s) + \lambda I & i=1 \\ \sum_{s=1}^{\ell-1}\sum_{j=1}^{M_s} \phi_{jk^*}(s)\ \phi'_{jk^*}(s) + \sum_{j=1}^{i-1} \phi_{jk^*}(\ell)\ \phi'_{jk^*}(\ell) + \lambda I & i \geq 2 \end{cases}$ is the design matrix.

Let $\mathcal{H}_{i\ell}^0$ be a sigma algebra generated by the feature vectors and the noise values upon the arrival of the i^{th} customer on day ℓ. Note that $\xi_{ik}(\ell)$ is σ_ξ-sub-Gaussian and the sequence $\{\sum_{s=1}^{\ell-1}\sum_{j=1}^{M_s} \xi_{jk^*}(s)\ \phi_{jk^*}(s) + \big(\sum_{j=1}^{i-1} \xi_{jk^*}(\ell)\ \phi_{jk^*}(\ell)\big)\mathbb{1}_{i\neq 1}\}_{\ell\in\mathcal{L}, i\in\mathcal{M}_\ell}$ is a *martingale* adapted to $\{\mathcal{H}_{i\ell}^0\}_{\ell\in\mathcal{L}, i\in\mathcal{M}_\ell}$. In the literature, it is proved that this martingale stays close to zero (see Theorem 1 in [1]). Accordingly, for any i, ℓ, and $\delta > 0$, the following holds with probability at least $1-\delta$:

$$\left\| \sum_{s=1}^{\ell-1}\sum_{j=1}^{M_s} \xi_{jk^*}(s)\ \phi_{jk^*}(s) + \big(\sum_{j=1}^{i-1} \xi_{jk^*}(\ell)\ \phi_{jk^*}(\ell)\big)\mathbb{1}_{i\neq 1} \right\|_{V_{i\ell}^{-1}} \leq \sigma_\xi \sqrt{2\log\left(\frac{\det(V_{i\ell})^{1/2}\det(\lambda I)^{-1/2}}{\delta}\right)}.$$

Term (II): First, we calculate the difference between the two estimators as:

$$\begin{aligned}
&\hat{w}_i^I(\ell) - \hat{w}_i(\ell) \\
&= V_{i\ell}^{-1}\Big(\sum_{(s,j,k^*)\in\{\mathcal{RF}^{(\mathcal{Q})}(i,\ell)\,\cup\,\mathcal{UF}^{(\mathcal{Q})}(i,\ell)\}} \mathcal{Q}_{jk^*}(s)\;\phi_{jk^*}(s) - \sum_{(s,j,k^*)\in\mathcal{RF}^{(\mathcal{Q})}(i,\ell)} \mathcal{Q}_{jk^*}(s)\;\phi_{jk^*}(s)\Big) \\
&= V_{i\ell}^{-1} \sum_{(s,j,k^*)\in\mathcal{UF}^{(\mathcal{Q})}(i,\ell)} \mathcal{Q}_{jk^*}(s)\;\phi_{jk^*}(s).
\end{aligned}$$

Accordingly, we have the following result:

$$\begin{aligned}
\langle\, \phi_{ik^*}(\ell), \hat{w}_i^I(\ell) - \hat{w}_i(\ell)\rangle &= \big\langle\, \phi_{ik^*}(\ell), V_{i\ell}^{-1} \sum_{(s,j,k^*)\in\mathcal{UF}^{(\mathcal{Q})}(i,\ell)} \mathcal{Q}_{jk^*}(s)\;\phi_{jk^*}(s)\big\rangle \\
&= \big\langle\, \phi_{ik^*}(\ell), \sum_{(s,j,k^*)\in\mathcal{UF}^{(\mathcal{Q})}(i,\ell)} \mathcal{Q}_{jk^*}(s)\;\phi_{jk^*}(s)\big\rangle_{V_{i\ell}^{-1}} \\
&\leq \|\phi_{ik^*}(\ell)\|_{V_{i\ell}^{-1}} \Bigg\| \sum_{(s,j,k^*)\in\mathcal{UF}^{(\mathcal{Q})}(i,\ell)} \mathcal{Q}_{jk^*}(s)\;\phi_{jk^*}(s) \Bigg\|_{V_{i\ell}^{-1}} \\
&\leq \|\phi_{ik^*}(\ell)\|_{V_{i\ell}^{-1}} \sum_{(s,j,k^*)\in\mathcal{UF}^{(\mathcal{Q})}(i,\ell)} \|\mathcal{Q}_{jk^*}(s)\;\phi_{jk^*}(s)\|_{V_{i\ell}^{-1}} \\
&\leq c_Q\, \|\phi_{ik^*}(\ell)\|_{V_{i\ell}^{-1}} \sum_{(s,j,k^*)\in\mathcal{UF}^{(\mathcal{Q})}(i,\ell)} \|\phi_{jk^*}(s)\|_{V_{i\ell}^{-1}}\,, \qquad (2.17)
\end{aligned}$$

where the first inequality is obtained using the Cauchy-Schwartz inequality $\langle a, b\rangle_M \leq \|a\|_M \; \|b\|_M$ for any vectors $a, b \in \mathbb{R}^n$ and matrix M. The second inequality holds by the triangle inequality. The last inequality holds because $\mathcal{Q}_{ik}(\ell) \in [0, c_Q]$.

Let Δ be the maximum number of days required for a feedback to be realized. Thus, on day ℓ, we know that the feedback outcomes of all customers who arrived before day $\ell - \Delta$ are realized for sure. Then, it is not hard to see that the summation of weighted norms in (2.17) can be upper bounded as:

$$\sum_{(s,j,k^*)\in\mathcal{UF}^{(\mathcal{Q})}(i,\ell)} \|\phi_{jk^*}(s)\|_{V_{i\ell}^{-1}} \leq \sum_{s=\max\{1,\ell-\Delta\}}^{\ell-1} \sum_{j=1}^{M_s} \|\phi_{jk^*}(s)\|_{V_{i\ell}^{-1}} + \big(\sum_{j=1}^{i-1} \|\phi_{jk^*}(\ell)\|_{V_{i\ell}^{-1}}\big)\mathbb{1}_{i\neq 1}.$$

Combining the bounds for Terms (I) and (II), we obtain the following:

$$\begin{aligned}
&\big|\langle\, \phi_{ik^*}(\ell), w\rangle - \langle\, \phi_{ik^*}(\ell), \hat{w}_i(\ell)\rangle\big| \\
&\leq \|\phi_{ik^*}(\ell)\|_{V_{i\ell}^{-1}} \left(\sigma_\xi \sqrt{2\log\left(\frac{\det(V_{i\ell})^{1/2}\det(\lambda I)^{-1/2}}{\delta}\right)} + \Sigma_{ik^*}^{(w)}(\ell)\right),
\end{aligned}$$

where $\Sigma_{ik^*}^{(w)}(\ell) = c_Q\Big(\sum_{s=\max\{1,\ell-\Delta\}}^{\ell-1} \sum_{j=1}^{M_s} \|\phi_{jk^*}(s)\|_{V_{i\ell}^{-1}} + \big(\sum_{j=1}^{i-1} \|\phi_{jk^*}(\ell)\|_{V_{i\ell}^{-1}} \big) \mathbb{1}_{i\neq 1}\Big) + \lambda^{1/2}$.

□

Proposition II.2 (Contextual Learning Loss Associated with Stochastic Reward in PRA-LD). *For any $\delta > 0$, the following holds with probability at least $1-\delta$.*

$$\begin{aligned}
&\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K} \mathbb{E}\big[r_{ik}(\ell)\, x_{ik}^{Aux^*}(\ell) - r_{ik}(\ell)\, x_{ik}^{Alg^*}(\ell)\big] \\
&\leq 2c_s\sqrt{N_L}\sqrt{2\,(d+K)\log\left(1+\tfrac{c_\phi^2 N_L}{\lambda(d+K)}\right)}\left(\sigma_\xi\sqrt{(d+K)\log\left(1+\tfrac{c_\phi^2 N_L}{\lambda(d+K)}\right)+\log\left(\tfrac{1}{\delta^2}\right)} + \lambda^{1/2}\right) \\
&+ 4\, c_s c_Q (d+K)\bar{M}(1+\Delta)\log\left(1+\frac{c_\phi^2 N_L}{\lambda(d+K)}\right) + c_s c_Q N_L \delta\ .
\end{aligned}$$

Proof. In the PRA-LD algorithm, we specify a prior distribution over the unknown model parameter w, and then update the posterior distribution by receiving new information. Upon arrival of a new customer, we take a random sample $\tilde{w}_i(\ell)$ from the posterior distribution over w to obtain the sample match quality. Recall that the history $\mathcal{H}_{i\ell}$ upon the arrival of the i^{th} customer on day ℓ includes context vectors, actions, and realized feedback outcomes. Since $\tilde{w}_i(\ell)$ is sampled from the posterior distribution $\mathbb{P}(w|\mathcal{H}_{i\ell})$, parameters w and $\tilde{w}_i(\ell)$ are *identically distributed* conditional on the history $\mathcal{H}_{i\ell}$, i.e., $\mathbb{P}(\tilde{w}_i(\ell)|\mathcal{H}_{i\ell}) = \mathbb{P}(w|\mathcal{H}_{i\ell})$.

Recall that $x_i^{Alg^*}(\ell) = \{x_{ik}^{Alg^*}(\ell)\}_{k\in\mathcal{K}}$ is the optimal solution of PRA-LD, and $x_i^{Aux^*}(\ell) = \{x_{ik}^{Aux^*}(\ell)\}_{k\in\mathcal{K}}$ is the optimal solution of the resource allocation mechanism to solve the auxiliary problem in which w is known in advance. In our proof techniques, we need to show that $x_i^{Alg^*}(\ell)$ and $x_i^{Aux^*}(\ell)$ are *identically distributed* conditional on the history $\mathcal{H}_{i\ell}$, i.e., $\mathbb{P}(x_i^{Alg^*}(\ell)|\mathcal{H}_{i\ell}) = \mathbb{P}(x_i^{Aux^*}(\ell)|\mathcal{H}_{i\ell})$. In the PRA-LD algorithm with a general resource allocation mechanism, assigning the i^{th} customer on day ℓ to a resource depends on the sample match quality $\tilde{q}_{ik}(\ell)$, expected service time $s_{ik}(\ell)$, and a variable used in the mechanism to keep track of the remaining capacity $n_{ik}^{Alg}(\ell)$. Similarly, this assignment decision in the auxiliary problem depends on $q_{ik}(\ell)$, $s_{ik}(\ell)$, and $n_{ik}^{Aux}(\ell)$. Accordingly, we need to argue that $\mathbb{P}(\tilde{q}_{ik}(\ell), n_{ik}^{Alg}(\ell)|\mathcal{H}_{i\ell}) = \mathbb{P}(q_{ik}(\ell), n_{ik}^{Aux}(\ell)|\mathcal{H}_{i\ell})$. First, note that $\tilde{q}_{ik}(\ell)$ and $n_{ik}^{Alg}(\ell)$ are independent random variables given $\mathcal{H}_{i\ell}$ because $n_{ik}^{Alg}(\ell)$ depends on all actions, context vectors, sample match quality values, and expected resource consumption values corresponding to the customers arrived prior to the i^{th} customer on day ℓ. The same argument holds for the auxiliary problem. Second, since w and $\tilde{w}_i(\ell)$ are identically distributed given the history $\mathcal{H}_{i\ell}$, we have $\mathbb{P}(\tilde{q}_{ik}(\ell)|\mathcal{H}_{i\ell}) = \mathbb{P}(q_{ik}(\ell)|\mathcal{H}_{i\ell})$, and $\mathbb{P}(n_{ik}^{Alg}(\ell)|\mathcal{H}_{i\ell}) = \mathbb{P}(n_{ik}^{Aux}(\ell)|\mathcal{H}_{i\ell})$. Thus, we conclude that $x_i^{Alg^*}(\ell)$ and $x_i^{Aux^*}(\ell)$ are *identically distributed* conditional on the history $\mathcal{H}_{i\ell}$.

Let $UB_{ik}^{(w)}(\ell)$ and $LB_{ik}^{(w)}(\ell)$ be the sequences of real-valued functions of $\mathcal{H}_{i\ell}$ and feature vector $\phi_{ik}(\ell)$ which are defined as:

$$UB_{ik}^{(w)}(\ell) = \min\Big\{c_Q,\ \max_{w\in\Theta_{i\ell}^{(w)}} \langle\, \phi_{ik}(\ell), w\rangle\Big\},\quad LB_{ik}^{(w)}(\ell) = \max\Big\{0,\ \min_{w\in\Theta_{i\ell}^{(w)}} \langle\, \phi_{ik}(\ell), w\rangle\Big\},$$

where $\Theta_{i\ell}^{(w)}$ is the confidence set that contains w with high probability. The above quantities indicate the largest and smallest possible values for the expected match quality given the history $\mathcal{H}_{i\ell}$, respectively.

Recall that $r_{ik}(\ell) = q_{ik}(\ell)\ s_{ik}(\ell)$. Since $s_{ik}(\ell) \le c_s$ is known, we have:

$$\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K} \mathbb{E}\big[r_{ik}(\ell)\ x_{ik}^{Aux^*}(\ell) - r_{ik}(\ell)\ x_{ik}^{Alg^*}(\ell)\big] \le$$
$$c_s \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K} \mathbb{E}\big[q_{ik}(\ell)\ x_{ik}^{Aux^*}(\ell) - q_{ik}(\ell)\ x_{ik}^{Alg^*}(\ell)\big].$$

Now we are ready to establish the following decomposition:

$$\begin{aligned}
&\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K} \mathbb{E}\big[q_{ik}(\ell)\ x_{ik}^{Aux^*}(\ell) - q_{ik}(\ell)\ x_{ik}^{Alg^*}(\ell)\big] \\
&= \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K} \mathbb{E}\left[\mathbb{E}\big[q_{ik}(\ell)\ x_{ik}^{Aux^*}(\ell) - q_{ik}(\ell)\ x_{ik}^{Alg^*}(\ell)|\mathcal{H}_{i\ell}\big]\right] \\
&= \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K} \mathbb{E}\Big[\mathbb{E}\big[q_{ik}(\ell)\ x_{ik}^{Aux^*}(\ell) + UB_{ik}^{(w)}(\ell)\ x_{ik}^{Alg^*}(\ell) \\
&\qquad\qquad - UB_{ik}^{(w)}(\ell)\ x_{ik}^{Aux^*}(\ell) - q_{ik}(\ell)\ x_{ik}^{Alg^*}(\ell)|\mathcal{H}_{i\ell}\big]\Big] \\
&= \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K} \mathbb{E}\big[UB_{ik}^{(w)}(\ell)\ x_{ik}^{Alg^*}(\ell) - q_{ik}(\ell)\ x_{ik}^{Alg^*}(\ell)\big] \\
&+ \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K} \mathbb{E}\big[q_{ik}(\ell)\ x_{ik}^{Aux^*}(\ell) - UB_{ik}^{(w)}(\ell)\ x_{ik}^{Aux^*}(\ell)\big].
\end{aligned} \tag{2.18}$$

Note that the above decomposition, first introduced by [95], is used to leverage the connection between PS-based and UCB algorithms. The first equality holds by the law of total expectation. The second equality holds by $\mathbb{P}(x_i^{Aux^*}(\ell)|\mathcal{H}_{i\ell}) = \mathbb{P}(x_i^{Alg^*}(\ell)|\mathcal{H}_{i\ell})$ and knowing that $UB_{ik}^{(w)}(\ell)$ is a deterministic function given the history $\mathcal{H}_{i\ell}$.

According to Proposition II.1, the following confidence bound holds with probability at least $1-\delta$:

$$\begin{aligned}&\big|\langle\, \phi_{ik^*}(\ell), w\rangle - \langle\, \phi_{ik^*}(\ell), \hat{w}_i(\ell)\rangle\big| \\ &\leq \|\phi_{ik^*}(\ell)\|_{V_{i\ell}^{-1}} \left(\sigma_\xi \sqrt{2\log\left(\frac{\det(V_{i\ell})^{1/2}\det(\lambda I)^{-1/2}}{\delta}\right)} + \Sigma_{ik^*}^{(w)}(\ell)\right),\end{aligned}$$

where $\Sigma_{ik^*}^{(w)}(\ell) = c_Q\Big(\sum_{s=\max\{1,\ell-\Delta\}}^{\ell-1}\sum_{j=1}^{M_s}\|\phi_{jk^*}(s)\|_{V_{i\ell}^{-1}} + \big(\sum_{j=1}^{i-1}\|\phi_{jk^*}(\ell)\|_{V_{i\ell}^{-1}}\big)\mathbb{1}_{i\neq 1}\Big) + \lambda^{1/2}$.

Accordingly, the two terms in (2.18) can be bounded with probability at least $1-\delta$.

For the first term in (2.18), we have:

$$\begin{aligned}&\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\mathbb{E}\big[UB_{ik}^{(w)}(\ell)\, x_{ik}^{Alg^*}(\ell) - q_{ik}(\ell)\, x_{ik}^{Alg^*}(\ell)\big] \\ &\leq \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\mathbb{E}\big[UB_{ik}^{(w)}(\ell)\, x_{ik}^{Alg^*}(\ell) - LB_{ik}^{(w)}(\ell)\, x_{ik}^{Alg^*}(\ell)\big].\end{aligned}$$

By Lemma II.2, the above term is bounded as:

$$\begin{aligned}&\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\mathbb{E}\big[UB_{ik}^{(w)}(\ell)\, x_{ik}^{Alg^*}(\ell) - LB_{ik}^{(w)}(\ell)\, x_{ik}^{Alg^*}(\ell)\big] \\ &\leq 2\sqrt{N_L}\sqrt{2\,(d+K)\log\left(1+\tfrac{c_\phi^2 N_L}{\lambda(d+K)}\right)}\left(\sigma_\xi\sqrt{(d+K)\log\left(1+\tfrac{c_\phi^2 N_L}{\lambda(d+K)}\right)+\log\left(\tfrac{1}{\delta^2}\right)} + \lambda^{1/2}\right) \\ &+ 4\, c_Q(d+K)\bar{M}(1+\Delta)\log\left(1+\frac{c_\phi^2 N_L}{\lambda(d+K)}\right).\end{aligned}$$

By Lemma II.3, the second term in (2.18) is bounded as:

$$\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\mathbb{E}\big[q_{ik}(\ell)\, x_{ik}^{Aux^*}(\ell) - UB_{ik}^{(w)}(\ell)\, x_{ik}^{Aux^*}(\ell)\big] \leq c_Q N_L \delta.$$

Putting the last two results together completes the proof. □

Proposition II.3 (Penalty Loss for Exceeding Resource Capacity). *For any $\delta > 0$, the following holds with probability at least $1 - \delta$.*

$$\mathbb{E}[\textsc{PenaltyLoss-I}] \leq c_Q \sigma_\eta K \sqrt{2N_L \log\left(\frac{2}{\delta}\right)} \, .$$

Proof. Recall that the resource consumption by a customer is stochastic in our problem and it is realized after the customer is assigned to a resource. Our online algorithm enforces the capacity constraints to hold only in *expectation*, where the algorithm assigns a customer to a resource by relying on the expected resource consumption value. Accordingly, the total realized resource consumption values of all customers assigned to a resource may exceed the capacity of that resource. Thus, we consider a *penalty* on the amount of capacity allocated in excess of the resource capacity. The expected penalty loss can be calculated as follows:

$$\mathbb{E}[\textsc{PenaltyLoss-I}] = c_Q \sum_{k=1}^{K} \mathbb{E}\left[\left(\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \mathcal{S}_{ik}(\ell)\, x_{ik}^{Alg^*}(\ell) - \mathcal{C}_k\right)^+\right],$$

where c_Q is the maximum possible value for the match quality.

To establish an upper bound for the above term, we decompose it into two terms:

$$\begin{aligned}
&\sum_{k=1}^{K} \mathbb{E}\left[\left(\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \mathcal{S}_{ik}(\ell)\, x_{ik}^{Alg^*}(\ell) - \mathcal{C}_k\right)^+\right] \\
&= \sum_{k=1}^{K} \mathbb{E}\left[\left(\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \left(\mathcal{S}_{ik}(\ell)\, x_{ik}^{Alg^*}(\ell) - s_{ik}(\ell)\, x_{ik}^{Alg^*}(\ell) + s_{ik}(\ell)\, x_{ik}^{Alg^*}(\ell)\right) - \mathcal{C}_k\right)^+\right] \\
&\leq \underbrace{\sum_{k=1}^{K} \mathbb{E}\left[\left|\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \left(\mathcal{S}_{ik}(\ell)\, x_{ik}^{Alg^*}(\ell) - s_{ik}(\ell)\, x_{ik}^{Alg^*}(\ell)\right)\right|\right]}_{\text{(I)}} \\
&+ \underbrace{\sum_{k=1}^{K} \mathbb{E}\left[\left(\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} s_{ik}(\ell)\, x_{ik}^{Alg^*}(\ell) - \mathcal{C}_k\right)^+\right]}_{\text{(II)}} .
\end{aligned}$$

The rest of the proof can be done by bounding each term, separately.

Term (I): First, we have:

$$\begin{aligned}
&\sum_{k=1}^{K} \mathbb{E}\left[\left|\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\Big(\mathcal{S}_{ik}(\ell)\, x_{ik}^{Alg^*}(\ell) - s_{ik}(\ell)\, x_{ik}^{Alg^*}(\ell)\Big)\right|\right] \\
&= \sum_{k=1}^{K} \mathbb{E}\left[\left|\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\Big(\mathcal{S}_{ik}(\ell) - s_{ik}(\ell)\Big) x_{ik}^{Alg^*}(\ell)\right|\right] \\
&\leq \sum_{k=1}^{K} \mathbb{E}\left[\left|\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\Big(\mathcal{S}_{ik}(\ell) - s_{ik}(\ell)\Big)\right|\right].
\end{aligned}$$

Note that $\mathcal{S}_{ik}(\ell) - s_{ik}(\ell)$ is the noise value which is a σ_η-sub-Gaussian random variable. By the Azuma-Hoeffding inequality for sub-Gaussian random variables and its corollary (see Lemma II.4), the following high-probability bound holds:

$$\left|\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\left(\mathcal{S}_{ik}(\ell) - s_{ik}(\ell)\right)\right| \leq \sigma_\eta \sqrt{2N_L \log\left(\frac{2}{\delta}\right)}, \quad \text{with probability at least } 1-\delta. \tag{2.19}$$

By (2.19), we have:

$$\sum_{k=1}^{K} \mathbb{E}\left[\left|\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\left(\mathcal{S}_{ik}(\ell) - s_{ik}(\ell)\right)\right|\right] \leq \ \sigma_\eta K \sqrt{2N_L \log\left(\frac{2}{\delta}\right)}.$$

Term (II): The algorithm guarantees that the capacity constraints hold in expectation. Then, we have:

$$\mathcal{C}_k \geq \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} s_{ik}(\ell)\, x_{ik}^{Alg^*}(\ell), \quad \forall\, k \in \mathcal{K}.$$

Accordingly, we have:

$$\sum_{k=1}^{K} \mathbb{E}\left[\left(\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} s_{ik}(\ell)\, x_{ik}^{Alg^*}(\ell) - \mathcal{C}_k\right)^{+}\right] \leq 0.$$

Summing the bounds obtained for Terms (I) and (II) completes the proof. □

Proposition II.4 (**Penalty Loss with Unknown Mean Resource Consumption**)**.** *For any $\delta > 0$, the following holds with probability at least $1 - 2\delta$.*

$$
\begin{aligned}
&\mathbb{E}[\textsc{PenaltyLoss-I}] \\
&\le c_Q \Bigg(2\sqrt{N_L}\sqrt{2\,(d+K)\log\left(1+\tfrac{c_\phi^2 N_L}{\lambda(d+K)}\right)}\Bigg(\sigma_\eta\sqrt{(d+K)\log\left(1+\tfrac{c_\phi^2 N_L}{\lambda(d+K)}\right)+\log\left(\tfrac{1}{\delta^2}\right)}+\lambda^{1/2}\Bigg) \\
&+ 4\,c_S(d+K)\bar{M}(1+\Delta)\log\left(1+\frac{c_\phi^2 N_L}{\lambda(d+K)}\right) + \sigma_\eta K\sqrt{2N_L\log\left(\frac{2}{\delta}\right)} + 2\,c_S N_L\delta\Bigg).
\end{aligned}
$$

Proof. Recall that if resource $k \in \mathcal{K}$ is chosen for the i^{th} customer on day ℓ, the customer uses $\mathcal{S}_{ik}(\ell)$ units of this resource, regardless of ℓ. We assume that $\mathcal{S}_{ik}(\ell) \in [\underline{c}_S, c_S]$ is a stochastic resource consumption following a linear model with the expected value:

$$\mathbb{E}[\mathcal{S}_{ik}(\ell)] = s_{ik}(\ell) = \langle\, \phi_{ik}(\ell), z\rangle,$$

where $z \in \mathbb{R}^{d+K}$ is the *unknown* model parameter. The noise values, $\eta_{ik}(\ell) = \mathcal{S}_{ik}(\ell) - \langle\, \phi_{ik}(\ell), z\rangle$, are independent σ_η-sub-Gaussian random variables.

According to Proposition II.3, the expected penalty loss can be calculated as follows:

$$\mathbb{E}[\textsc{PenaltyLoss-I}] = c_Q\ \sum_{k=1}^{K}\mathbb{E}\left[\left(\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\mathcal{S}_{ik}(\ell)\ x_{ik}^{Alg^*}(\ell) - \mathcal{C}_k\right)^{+}\right],$$

where c_Q is the maximum possible value for the match quality.

To establish an upper bound for the above term, we decompose it into two terms:

$$
\begin{aligned}
&\sum_{k=1}^{K}\mathbb{E}\left[\left(\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\mathcal{S}_{ik}(\ell)\ x_{ik}^{Alg^*}(\ell) - \mathcal{C}_k\right)^{+}\right] \\
&\le \underbrace{\sum_{k=1}^{K}\mathbb{E}\left[\left|\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\left(\mathcal{S}_{ik}(\ell)\ x_{ik}^{Alg^*}(\ell) - s_{ik}(\ell)\ x_{ik}^{Alg^*}(\ell)\right)\right|\right]}_{\text{(I)}} \\
&+ \underbrace{\sum_{k=1}^{K}\mathbb{E}\left[\left(\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}s_{ik}(\ell)\ x_{ik}^{Alg^*}(\ell) - \mathcal{C}_k\right)^{+}\right]}_{\text{(II)}}.
\end{aligned}
$$

The rest of the proof can be done by bounding each term, separately.

Term (I): Similar to Proposition II.3, with probability at least $1 - \delta$, the first term is bounded as follows:

$$\sum_{k=1}^{K} \mathbb{E}\left[\left|\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\Big(\mathcal{S}_{ik}(\ell)\, x_{ik}^{Alg^*}(\ell) - s_{ik}(\ell)\, x_{ik}^{Alg^*}(\ell)\Big)\right|\right] \leq\ \sigma_\eta K\sqrt{2N_L \log\Big(\frac{2}{\delta}\Big)}.$$

Term (II): We define $\tilde{s}_{ik}(\ell) = \langle\, \phi_{ik}(\ell), \tilde{z}_i(\ell)\rangle$, where $\tilde{z}_i(\ell)$ is sampled from the posterior distribution $\mathbb{P}(z|\mathcal{H}_{i\ell})$. The algorithm guarantees that the capacity constraints hold in expectation. Hence:

$$\mathcal{C}_k \geq\ \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \tilde{s}_{ik}(\ell)\, x_{ik}^{Alg^*}(\ell), \quad \forall\, k \in \mathcal{K}.$$

Accordingly, we have:

$$\begin{aligned}
&\sum_{k=1}^{K}\mathbb{E}\left[\left(\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} s_{ik}(\ell)\, x_{ik}^{Alg^*}(\ell) - \mathcal{C}_k\right)^+\right] \\
&\leq\ \sum_{k=1}^{K}\mathbb{E}\left[\left(\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\Big(s_{ik}(\ell)\, x_{ik}^{Alg^*}(\ell) - \tilde{s}_{ik}(\ell)\, x_{ik}^{Alg^*}(\ell)\Big)\right)^+\right] \\
&=\ \sum_{k=1}^{K}\mathbb{E}\left[\left(\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\Big(s_{ik}(\ell) - \tilde{s}_{ik}(\ell)\Big)x_{ik}^{Alg^*}(\ell)\right)^+\right] \\
&\leq\ \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\mathbb{E}\left[\Big(s_{ik}(\ell) - \tilde{s}_{ik}(\ell)\Big)^+ x_{ik}^{Alg^*}(\ell)\right],
\end{aligned}$$

where the last inequality holds by $(\sum_{i=1}^n a_i)^+ \leq \sum_{i=1}^n (a_i)^+$ for any $a_i \in \mathbb{R}$.

Let $UB_{ik}^{(z)}(\ell)$ and $LB_{ik}^{(z)}(\ell)$ be the sequences of real-valued functions of $\mathcal{H}_{i\ell}$ and feature vector $\phi_{ik}(\ell)$ which are defined as:

$$UB_{ik}^{(z)}(\ell) = \min\Big\{c_S,\ \max_{z\in\Theta_{i\ell}^{(z)}} \langle\, \phi_{ik}(\ell), z\rangle\Big\}, \quad LB_{ik}^{(z)}(\ell) = \max\Big\{\underline{c}_S,\ \min_{z\in\Theta_{i\ell}^{(z)}} \langle\, \phi_{ik}(\ell), z\rangle\Big\},$$

where $\Theta_{i\ell}^{(z)}$ is the confidence set that contains z with high probability. The above quantities indicate the largest and smallest possible values for the expected resource consumption given the history $\mathcal{H}_{i\ell}$, respectively.

Next, we establish the following decomposition:

$$\begin{aligned}
&\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\mathbb{E}\left[\left(s_{ik}(\ell)-\tilde{s}_{ik}(\ell)\right)^{+}x_{ik}^{Alg^*}(\ell)\right]\\
&=\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\mathbb{E}\left[\left(s_{ik}(\ell)-UB_{ik}^{(z)}(\ell)+UB_{ik}^{(z)}(\ell)-LB_{ik}^{(z)}(\ell)+LB_{ik}^{(z)}(\ell)-\tilde{s}_{ik}(\ell)\right)^{+}x_{ik}^{Alg^*}(\ell)\right]\\
&\leq\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\mathbb{E}\left[\left(s_{ik}(\ell)-UB_{ik}^{(z)}(\ell)\right)^{+}x_{ik}^{Alg^*}(\ell)\right]+\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\mathbb{E}\left[\left(UB_{ik}^{(z)}(\ell)-LB_{ik}^{(z)}(\ell)\right)x_{ik}^{Alg^*}(\ell)\right]\\
&+\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\mathbb{E}\left[\left(LB_{ik}^{(z)}(\ell)-\tilde{s}_{ik}(\ell)\right)^{+}x_{ik}^{Alg^*}(\ell)\right],
\end{aligned}$$

where the inequality holds by $(a+b+c)^+\leq a^++b^++c^+$ for any $a,b,c\in\mathbb{R}$ and the fact that $UB_{ik}^{(z)}(\ell)\geq LB_{ik}^{(z)}(\ell)$ for any $\ell\in\mathcal{L}, i\in\mathcal{M}_\ell, k\in\mathcal{K}$.

Using the confidence bound established in Proposition II.1, the above three terms can be bounded with probability at least $1-\delta$.

By Lemma II.3, the first term can be bounded as:

$$\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\mathbb{E}\left[\left(s_{ik}(\ell)-UB_{ik}^{(z)}(\ell)\right)^{+}x_{ik}^{Alg^*}(\ell)\right]\leq c_S N_L\delta.$$

By Lemma II.2, the second term can be bounded as:

$$\begin{aligned}
&\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\mathbb{E}\left[\left(UB_{ik}^{(z)}(\ell)-LB_{ik}^{(z)}(\ell)\right)x_{ik}^{Alg^*}(\ell)\right]\\
&\leq 2\sqrt{N_L}\sqrt{2\,(d+K)\log\left(1+\tfrac{c_\phi^2N_L}{\lambda(d+K)}\right)}\left(\sigma_\eta\sqrt{(d+K)\log\left(1+\tfrac{c_\phi^2N_L}{\lambda(d+K)}\right)+\log\left(\tfrac{1}{\delta^2}\right)}+\lambda^{1/2}\right)\\
&+4\,c_S(d+K)\bar{M}(1+\Delta)\log\left(1+\frac{c_\phi^2N_L}{\lambda(d+K)}\right),
\end{aligned}$$

where λ is the regularization parameter. Note that assuming the same regularization parameter when estimating w and z is only to keep the notation simple and it is not necessary.

For the third term, we have the following because we cannot assign a customer to more than one resource:

$$\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\mathbb{E}\left[\left(LB_{ik}^{(z)}(\ell)-\tilde{s}_{ik}(\ell)\right)^{+}x_{ik}^{Alg^*}(\ell)\right]=\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\mathbb{E}\left[\left(LB_{ik}^{(z)}(\ell)-\tilde{s}_{ik}(\ell)\right)^{+}\mathbb{1}_{k(i,\ell)=k}\right]$$
$$=\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\mathbb{E}\left[\left(LB_{ik^*}^{(z)}(\ell)-\tilde{s}_{ik^*}(\ell)\right)^{+}\right].$$

Since parameters z and $\tilde{z}_i(\ell)$ are *identically distributed* conditional on the history $\mathcal{H}_{i\ell}$ (i.e., $\mathbb{P}(\tilde{z}_i(\ell)|\mathcal{H}_{i\ell})=\mathbb{P}(z|\mathcal{H}_{i\ell})$), we have $\mathbb{P}(\tilde{s}_{ik^*}(\ell)|\mathcal{H}_{i\ell})=\mathbb{P}(s_{ik^*}(\ell)|\mathcal{H}_{i\ell})$. Because $LB_{ik^*}^{(z)}(\ell)$ is deterministic given $\mathcal{H}_{i\ell}$, we have:

$$\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\mathbb{E}\left[\left(LB_{ik^*}^{(z)}(\ell)-\tilde{s}_{ik^*}(\ell)\right)^{+}\right]=\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\mathbb{E}\left[\mathbb{E}\Big[\left(LB_{ik^*}^{(z)}(\ell)-\tilde{s}_{ik^*}(\ell)\right)^{+}\Big|\mathcal{H}_{i\ell}\Big]\right]$$
$$=\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\mathbb{E}\left[\mathbb{E}\Big[\left(LB_{ik^*}^{(z)}(\ell)-s_{ik^*}(\ell)\right)^{+}\Big|\mathcal{H}_{i\ell}\Big]\right]$$
$$=\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\mathbb{E}\left[\left(LB_{ik^*}^{(z)}(\ell)-s_{ik^*}(\ell)\right)^{+}\right].$$

Accordingly, the third term can be bounded as:

$$\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\mathbb{E}\left[\left(LB_{ik^*}^{(z)}(\ell)-s_{ik^*}(\ell)\right)^{+}\right]\leq c_S N_L\delta,$$

where the upper bound is obtained by Lemma II.3.

Summing the bounds obtained for Terms (I) and (II) completes the proof. □

2.7.2 Appendix B. Technical Results for the PAS-LD Algorithm

Proposition II.5 (Confidence Bound for PAS-LD under Delayed Feedback). *For any i, ℓ, and $\delta>0$, the following holds with probability at least $1-2\delta$.*

$$\left|p\,\langle\,\phi_{ik^*t^*}(\ell),\ w\rangle-\langle\,\phi_{ik^*t^*}(\ell),\hat{w}_i^c(\ell)\rangle\right|$$
$$\leq\|\phi_{ik^*t^*}(\ell)\|_{V_{i\ell}^{-1}}\left((c_Q+\sigma_\xi)\sqrt{2\log\left(\frac{\det(V_{i\ell})^{1/2}\det(\lambda I)^{-1/2}}{\delta}\right)}+\Sigma_{ik^*t^*}^{(pw)}(\ell)\right),$$

where $V_{i\ell} = \begin{cases} \sum_{s=1}^{\ell-1}\sum_{j=1}^{M_s}\phi_{jk^*t^*}(s)\ \phi'_{jk^*t^*}(s) + \lambda I & i = 1 \\ \sum_{s=1}^{\ell-1}\sum_{j=1}^{M_s}\phi_{jk^*t^*}(s)\ \phi'_{jk^*t^*}(s) + \sum_{j=1}^{i-1}\phi_{jk^*t^*}(\ell)\ \phi'_{jk^*t^*}(\ell) + \lambda I & i \geq 2 \end{cases}$ *is the design matrix such that* $\lambda > 0$ *and*
$\Sigma_{ik^*t^*}^{(pw)}(\ell) = \ c_Q\Big(\sum_{s=\max\{1,\ell-\Delta\}}^{\ell-1}\sum_{j=1}^{M_s}\|\phi_{jk^*t^*}(s)\|_{V_{i\ell}^{-1}} + \big(\sum_{j=1}^{i-1}\|\phi_{jk^*t^*}(\ell)\|_{V_{i\ell}^{-1}}\big)\mathbb{1}_{i\neq 1}\Big) + \lambda^{1/2}.$

Proof. In our advance scheduling problem, customers may not show up on the service date after being assigned to a server and date. This adds an additional layer of complexity for estimating the parameter w, because the match quality feedback cannot be observed at all if the customer does not show up on the service date. In this proposition, we provide an estimator for pw and a confidence bound on the expected match quality by taking into account the probability that a customer shows up.

First, we define an ideal estimator of pw under no delayed feedback assumption as follows:

$$\hat{w}_i^{c-I}(\ell) = \left(\sum_{s=1}^{\ell-1}\sum_{j=1}^{M_s}\phi_{jk^*t^*}(s)\ \phi'_{jk^*t^*}(s) + \left(\sum_{j=1}^{i-1}\phi_{jk^*t^*}(\ell)\ \phi'_{jk^*t^*}(\ell)\right)\mathbb{1}_{i\neq 1} + \lambda I\right)^{-1}$$
$$\left(\sum_{s=1}^{\ell-1}\sum_{j=1}^{M_s}\mathcal{SU}_j(s)\mathcal{Q}_{jk^*t^*}(s)\ \phi_{jk^*t^*}(s) + \left(\sum_{j=1}^{i-1}\mathcal{SU}_j(\ell)\mathcal{Q}_{jk^*t^*}(\ell)\ \phi_{jk^*t^*}(\ell)\right)\mathbb{1}_{i\neq 1}\right),$$

where $\hat{w}_i^{c-I}(\ell)$ is obtained by *assuming* that all no-show and match quality feedback outcomes of prior customers are realized by the time we calculate $\hat{w}_i^{c-I}(\ell)$. Note that $\hat{w}_i^{c-I}(\ell)$ is different from $\hat{w}_i^{I}(\ell)$ defined in Proposition II.1 because it also captures the no-show behavior.

In our setting, neither the match quality feedback nor the no-show feedback is available right after scheduling a customer. In particular, some feedback outcomes needed for obtaining $\hat{w}_i^{c-I}(\ell)$ might not be available. Thus, $\hat{w}_i^{c-I}(\ell)$ cannot be used as an estimate for pw. Let (ℓ, i, k^*, t^*) be a tuple referring to the i^{th} customer on day ℓ assigned to server-date (k^*, t^*). We define $\mathcal{RF}^{(\mathcal{SU},\mathcal{Q})}(i,\ell)$ as the set containing tuples (s, j, k^*, t^*) of customers with *realized* no-show and match quality feedback outcomes before the arrival of the i^{th} customer on day ℓ, where $s \leq \ell$ and $j < i$. Similarly, let $\mathcal{UF}^{(\mathcal{SU},\mathcal{Q})}(i,\ell)$ be the set containing tuples (s, j, k^*, t^*) of customers with *unrealized* no-show and match quality feedback outcomes before the arrival of the i^{th} customer on day ℓ. We define $\hat{w}_i^{c}(\ell)$ as an estimator of pw under delayed feedback. This estimator uses only the *available* information up to the current time and can be obtained as follows:

$$\begin{aligned}
\hat{w}_i^c(\ell) = &\left(\sum_{s=1}^{\ell-1}\sum_{j=1}^{M_s} \phi_{jk^*t^*}(s)\ \phi^{'}_{jk^*t^*}(s) + \Big(\sum_{j=1}^{i-1} \phi_{jk^*t^*}(\ell)\ \phi^{'}_{jk^*t^*}(\ell)\Big)\mathbb{1}_{i\neq 1} + \lambda I\right)^{-1} \\
&\left(\sum_{(s,j,k^*,t^*)\in\mathcal{RF}^{(\mathcal{SU},\mathcal{Q})}(i,\ell)} \mathcal{SU}_j(s)\mathcal{Q}_{jk^*t^*}(s)\ \phi_{jk^*t^*}(s)\right).
\end{aligned}$$

We would like to derive an upper bound for $\big|p\ \langle\, \phi_{ik^*t^*}(\ell), w\rangle - \langle\, \phi_{ik^*t^*}(\ell), \hat{w}_i^c(\ell)\rangle\big|$. Using the triangle inequality, we have:

$$\begin{aligned}
&\big|p\ \langle\, \phi_{ik^*t^*}(\ell), w\rangle - \langle\, \phi_{ik^*t^*}(\ell), \hat{w}_i^c(\ell)\rangle\big| \\
&\leq \underbrace{\big|p\ \langle\, \phi_{ik^*t^*}(\ell), w\rangle - \langle\, \phi_{ik^*t^*}(\ell), \hat{w}_i^{c-I}(\ell)\rangle\big|}_{\text{(I)}} + \underbrace{\big|\langle\, \phi_{ik^*t^*}(\ell), \hat{w}_i^{c-I}(\ell)\rangle - \langle\, \phi_{ik^*t^*}(\ell), \hat{w}_i^c(\ell)\rangle\big|}_{\text{(II)}}.
\end{aligned}$$

We construct the confidence bound by bounding the two terms.

Term (I): We start with bounding $\big\|\hat{w}_i^{c-I}(\ell) - p\ w\big\|_{V_{i\ell}}$.

$$\begin{aligned}
&\big\|\hat{w}_i^{c-I}(\ell) - p\ w\big\|_{V_{i\ell}} \\
&= \left\|V_{i\ell}^{-1}\left(\sum_{s=1}^{\ell-1}\sum_{j=1}^{M_s} \mathcal{SU}_j(s)\mathcal{Q}_{jk^*t^*}(s)\ \phi_{jk^*t^*}(s) + \Big(\sum_{j=1}^{i-1} \mathcal{SU}_j(\ell)\mathcal{Q}_{jk^*t^*}(\ell)\ \phi_{jk^*t^*}(\ell)\Big)\mathbb{1}_{i\neq 1}\right) - p\ w\right\|_{V_{i\ell}} \\
&\leq \left\|\sum_{s=1}^{\ell-1}\sum_{j=1}^{M_s} \epsilon_j(s)\mathcal{Q}_{jk^*t^*}(s)\ \phi_{jk^*t^*}(s) + \Big(\sum_{j=1}^{i-1} \epsilon_j(\ell)\mathcal{Q}_{jk^*t^*}(\ell)\ \phi_{jk^*t^*}(\ell)\Big)\mathbb{1}_{i\neq 1}\right\|_{V_{i\ell}^{-1}} \\
&+ \left\|V_{i\ell}^{-1}\left(\sum_{s=1}^{\ell-1}\sum_{j=1}^{M_s} p\ \mathcal{Q}_{jk^*t^*}(s)\ \phi_{jk^*t^*}(s) + \Big(\sum_{j=1}^{i-1} p\ \mathcal{Q}_{jk^*t^*}(\ell)\ \phi_{jk^*t^*}(\ell)\Big)\mathbb{1}_{i\neq 1}\right) - p\ w\right\|_{V_{i\ell}} \\
&\leq \left\|\sum_{s=1}^{\ell-1}\sum_{j=1}^{M_s} \epsilon_j(s)\mathcal{Q}_{jk^*t^*}(s)\ \phi_{jk^*t^*}(s) + \Big(\sum_{j=1}^{i-1} \epsilon_j(\ell)\mathcal{Q}_{jk^*t^*}(\ell)\ \phi_{jk^*t^*}(\ell)\Big)\mathbb{1}_{i\neq 1}\right\|_{V_{i\ell}^{-1}} \\
&+ p\ \left\|\sum_{s=1}^{\ell-1}\sum_{j=1}^{M_s} \xi_{jk^*t^*}(s)\ \phi_{jk^*t^*}(s) + \Big(\sum_{j=1}^{i-1} \xi_{jk^*t^*}(\ell)\ \phi_{jk^*t^*}(\ell)\Big)\mathbb{1}_{i\neq 1}\right\|_{V_{i\ell}^{-1}} \\
&+ p\ \left\|V_{i\ell}^{-1}\left(\sum_{s=1}^{\ell-1}\sum_{j=1}^{M_s}\ \phi_{jk^*t^*}(s)\ \phi^{'}_{jk^*t^*}(s)\ w + \Big(\sum_{j=1}^{i-1}\ \phi_{jk^*t^*}(\ell)\ \phi^{'}_{jk^*t^*}(\ell)\ w\Big)\mathbb{1}_{i\neq 1}\right) - w\right\|_{V_{i\ell}},
\end{aligned} \tag{2.20}$$

where the first inequality is obtained by replacing $\mathcal{SU}_i(\ell)$ with $p+\epsilon_i(\ell)$ and using the triangle inequality. The last inequality is obtained by replacing $\mathcal{Q}_{ik^*t^*}(\ell)$ with $\langle\phi_{ik^*t^*}(\ell), w\rangle + \xi_{ik^*t^*}(\ell)$

in the second term and using the triangle inequality.

Let $\bar{\mathcal{H}}_{i\ell}^{0}$ be a sigma algebra generated by the feature vectors and the noise values upon the arrival of the i^{th} customer on day ℓ. Note that $\epsilon_i(\ell)\mathcal{Q}_{ikt}(\ell)$ is c_Q-sub-Gaussian and $\xi_{ikt}(\ell)$ is σ_ξ-sub-Gaussian. Then, we have two sequences that are martingales adapted to $\{\bar{\mathcal{H}}_{i\ell}^{0}\}_{\ell\in\mathcal{L},i\in\mathcal{M}_\ell}$. First one is $\{\sum_{s=1}^{\ell-1}\sum_{j=1}^{M_s}\epsilon_j(s)\mathcal{Q}_{jk^*t^*}(s)\ \phi_{jk^*t^*}(s)+(\sum_{j=1}^{i-1}\epsilon_j(\ell)\mathcal{Q}_{jk^*t^*}(\ell)\ \phi_{jk^*t^*}(\ell))\mathbb{1}_{i\neq1}\}_{\ell\in\mathcal{L},i\in\mathcal{M}_\ell}$ and the second one is $\{\sum_{s=1}^{\ell-1}\sum_{j=1}^{M_s}\xi_{jk^*t^*}(s)\ \phi_{jk^*t^*}(s)+(\sum_{j=1}^{i-1}\xi_{jk^*t^*}(\ell)\ \phi_{jk^*t^*}(\ell))\mathbb{1}_{i\neq1}\}_{\ell\in\mathcal{L},i\in\mathcal{M}_\ell}$. Then, the first and second terms in (2.20) can be bounded using the same technique used to bound Term (I) in Proposition II.1. Accordingly, for any i, ℓ, and $\delta > 0$, each of the followings holds with probability at least $1-\delta$:

$$\begin{aligned}
&\left\|\sum_{s=1}^{\ell-1}\sum_{j=1}^{M_s}\epsilon_j(s)\mathcal{Q}_{jk^*t^*}(s)\ \phi_{jk^*t^*}(s)+\Big(\sum_{j=1}^{i-1}\epsilon_j(\ell)\mathcal{Q}_{jk^*t^*}(\ell)\ \phi_{jk^*t^*}(\ell)\Big)\mathbb{1}_{i\neq1}\right\|_{V_{i\ell}^{-1}}\\
&\leq c_Q\sqrt{2\log\left(\frac{\det(V_{i\ell})^{1/2}\det(\lambda I)^{-1/2}}{\delta}\right)},\\
&p\ \left\|\sum_{s=1}^{\ell-1}\sum_{j=1}^{M_s}\xi_{jk^*t^*}(s)\ \phi_{jk^*t^*}(s)+\Big(\sum_{j=1}^{i-1}\xi_{jk^*t^*}(\ell)\ \phi_{jk^*t^*}(\ell)\Big)\mathbb{1}_{i\neq1}\right\|_{V_{i\ell}^{-1}}\\
&\leq \sigma_\xi\sqrt{2\log\left(\frac{\det(V_{i\ell})^{1/2}\det(\lambda I)^{-1/2}}{\delta}\right)}.
\end{aligned}$$

Next, we simplify the third term in (2.20). Let $W_{i\ell}=V_{i\ell}-\lambda I$. Then, we have:

$$\begin{aligned}
&p\ \left\|V_{i\ell}^{-1}\left(\sum_{s=1}^{\ell-1}\sum_{j=1}^{M_s}\ \phi_{jk^*t^*}(s)\ \phi'_{jk^*t^*}(s)w+\Big(\sum_{j=1}^{i-1}\ \phi_{jk^*t^*}(\ell)\ \phi'_{jk^*t^*}(\ell)\ w\Big)\mathbb{1}_{i\neq1}\right)-w\right\|_{V_{i\ell}}\\
&=p\ \left\|\big(V_{i\ell}^{-1}W_{i\ell}-I\big)w\right\|_{V_{i\ell}},\\
&=p\ \Big(w'(V_{i\ell}^{-1}W_{i\ell}-I)V_{i\ell}(V_{i\ell}^{-1}W_{i\ell}-I)w\Big)^{1/2}\\
&=p\ \Big(w'(I-V_{i\ell}^{-1}W_{i\ell})V_{i\ell}(I-V_{i\ell}^{-1}W_{i\ell})w\Big)^{1/2}\\
&=p\ \lambda^{1/2}\Big(w'(I-V_{i\ell}^{-1}W_{i\ell})w\Big)^{1/2}\\
&\leq p\ \lambda^{1/2}\ \|w\|\leq\lambda^{1/2},
\end{aligned}$$

where the inequality holds because $\|w\|\leq 1$.

Accordingly, Term (I) is bounded with probability at least $1-2\delta$:

$$\begin{aligned}
&\left|p\,\langle\,\phi_{ik^*t^*}(\ell), w\rangle - \langle\,\phi_{ik^*t^*}(\ell), \hat{w}_i^{c-I}(\ell)\rangle\right| \\
&\leq \|\phi_{ik^*t^*}(\ell)\|_{V_{i\ell}^{-1}} \left\|\hat{w}_i^{c-I}(\ell) - pw\right\|_{V_{i\ell}} \\
&\leq \|\phi_{ik^*t^*}(\ell)\|_{V_{i\ell}^{-1}} \left((c_Q+\sigma_\xi)\sqrt{2\log\left(\frac{\det(V_{i\ell})^{1/2}\det(\lambda I)^{-1/2}}{\delta}\right)} + \lambda^{1/2}\right).
\end{aligned}$$

Term (II): This term can be bounded following our technique used to bound Term (II) in Proposition II.1. Accordingly, we have:

$$\begin{aligned}
&\left|\langle\,\phi_{ik^*t^*}(\ell), \hat{w}_i^{c-I}(\ell)\rangle - \langle\,\phi_{ik^*t^*}(\ell), \hat{w}_i^{c}(\ell)\rangle\right| \\
&\leq\; c_Q\,\|\phi_{ik^*t^*}(\ell)\|_{V_{i\ell}^{-1}} \sum_{(s,j,k^*,t^*)\in\mathcal{UF}^{(\mathcal{SU},\mathcal{Q})}(i,\ell)} \|\phi_{jk^*t^*}(s)\|_{V_{i\ell}^{-1}} \\
&\leq c_Q\,\|\phi_{ik^*t^*}(\ell)\|_{V_{i\ell}^{-1}} \left(\sum_{s=\max\{1,\ell-\Delta\}}^{\ell-1}\sum_{j=1}^{M_s}\|\phi_{jk^*t^*}(s)\|_{V_{i\ell}^{-1}} + \left(\sum_{j=1}^{i-1}\|\phi_{jk^*t^*}(\ell)\|_{V_{i\ell}^{-1}}\right)\mathbb{1}_{i\neq 1}\right).
\end{aligned}$$

Combining the bounds for Terms (I) and (II), we obtain the following:

$$\begin{aligned}
&\left|p\,\langle\,\phi_{ik^*t^*}(\ell),\; w\rangle - \langle\,\phi_{ik^*t^*}(\ell), \hat{w}_i^{c}(\ell)\rangle\right| \\
&\leq \|\phi_{ik^*t^*}(\ell)\|_{V_{i\ell}^{-1}} \left((c_Q+\sigma_\xi)\sqrt{2\log\left(\frac{\det(V_{i\ell})^{1/2}\det(\lambda I)^{-1/2}}{\delta}\right)} + \Sigma_{ik^*t^*}^{(pw)}(\ell)\right),
\end{aligned}$$

where $\Sigma_{ik^*t^*}^{(pw)}(\ell) = c_Q\Big(\sum_{s=\max\{1,\ell-\Delta\}}^{\ell-1}\sum_{j=1}^{M_s}\|\phi_{jk^*t^*}(s)\|_{V_{i\ell}^{-1}} + \big(\sum_{j=1}^{i-1}\|\phi_{jk^*t^*}(\ell)\|_{V_{i\ell}^{-1}}\big)\mathbb{1}_{i\neq 1}\Big) + \lambda^{1/2}$. □

Proposition II.6 (Contextual Learning Loss Associated with Stochastic Reward in PAS-LD)**.** *For any $\delta>0$, the following holds with probability at least $1-2\delta$.*

$$\begin{aligned}
&\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\sum_{t=\ell+1}^{L}\mathbb{E}\big[r_{ikt}(\ell)\,x_{ikt}^{Aux^*}(\ell) - r_{ikt}(\ell)\,x_{ikt}^{Alg^*}(\ell)\big] \\
&\leq 2c_s\sqrt{N_L}\sqrt{2\,(d+A)\log\left(1+\tfrac{c_\phi^2N_L}{\lambda(d+A)}\right)}\left((c_Q+\sigma_\xi)\sqrt{(d+A)\log\left(1+\tfrac{c_\phi^2N_L}{\lambda(d+A)}\right)+\log\left(\tfrac{1}{\delta^2}\right)}+\lambda^{1/2}\right) \\
&+ 4\;c_sc_Q(d+A)\bar{M}(1+\Delta)\log\left(1+\frac{c_\phi^2N_L}{\lambda(d+A)}\right) + 2\;c_sc_QN_L\delta.
\end{aligned}$$

Proof. Recall that $x_i^{Alg^*}(\ell) = \{x_{ikt}^{Alg^*}(\ell)\}_{k\in\mathcal{K},t\geq\ell+1}$ is the optimal solution of the PAS-LD algorithm, and $x_i^{Aux^*}(\ell) = \{x_{ikt}^{Aux^*}(\ell)\}_{k\in\mathcal{K},t\geq\ell+1}$ is the optimal solution of the advance scheduling mechanism to solve the auxiliary problem in which the unknown model parameters (p and w) are known in advance. Similar to our arguments in Proposition II.2, $x_i^{Alg^*}(\ell)$ and $x_i^{Aux^*}(\ell)$ are *identically distributed* conditional on the history $\bar{\mathcal{H}}_{i\ell}$, i.e., $\mathbb{P}(x_i^{Alg^*}(\ell)|\bar{\mathcal{H}}_{i\ell}) = \mathbb{P}(x_i^{Aux^*}(\ell)|\bar{\mathcal{H}}_{i\ell})$.

Let $UB_{ikt}^{(pw)}(\ell)$ and $LB_{ikt}^{(pw)}(\ell)$ be the sequences of real-valued functions of $\bar{\mathcal{H}}_{i\ell}$ and feature vector $\phi_{ikt}(\ell)$ which are defined as:

$$UB_{ikt}^{(pw)}(\ell) = \min\Big\{c_Q,\ \max_{pw\in\Theta_{i\ell}^{(pw)}} \langle\, \phi_{ikt}(\ell), pw\rangle\Big\}, \quad LB_{ikt}^{(pw)}(\ell) = \max\Big\{0,\ \min_{pw\in\Theta_{i\ell}^{(pw)}} \langle\, \phi_{ikt}(\ell), pw\rangle\Big\},$$

where $\Theta_{i\ell}^{(pw)}$ is the confidence set that contains pw with high probability.

Recall that $r_{ikt}(\ell) = q_{ikt}(\ell)\ s_{ikt}(\ell)$. Since $s_{ikt}(\ell) \leq c_s$ is known, we have:

$$\begin{aligned}
&\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\sum_{t=\ell+1}^{L} \mathbb{E}\big[r_{ikt}(\ell)\ x_{ikt}^{Aux^*}(\ell) - r_{ikt}(\ell)\ x_{ikt}^{Alg^*}(\ell)\big] \\
&\leq c_s \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\sum_{t=\ell+1}^{L} \mathbb{E}\big[q_{ikt}(\ell)\ x_{ikt}^{Aux^*}(\ell) - q_{ikt}(\ell)\ x_{ikt}^{Alg^*}(\ell)\big].
\end{aligned}$$

Accordingly, we establish the following decomposition:

$$\begin{aligned}
&\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\sum_{t=\ell+1}^{L} \mathbb{E}\big[q_{ikt}(\ell)\ x_{ikt}^{Aux^*}(\ell) - q_{ikt}(\ell)\ x_{ikt}^{Alg^*}(\ell)\big] \\
&= \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\sum_{t=\ell+1}^{L} \mathbb{E}\Big[\mathbb{E}\big[q_{ikt}(\ell)\ x_{ikt}^{Aux^*}(\ell) - q_{ikt}(\ell)\ x_{ikt}^{Alg^*}(\ell)|\bar{\mathcal{H}}_{i\ell}\big]\Big] \\
&= \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\sum_{t=\ell+1}^{L} \mathbb{E}\Big[\mathbb{E}\big[q_{ikt}(\ell)\ x_{ikt}^{Aux^*}(\ell) + UB_{ikt}^{(pw)}(\ell)\ x_{ikt}^{Alg^*}(\ell) \\
&\qquad - UB_{ikt}^{(pw)}(\ell)\ x_{ikt}^{Aux^*}(\ell) - q_{ikt}(\ell)\ x_{ikt}^{Alg^*}(\ell)|\bar{\mathcal{H}}_{i\ell}\big]\Big] \\
&= \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\sum_{t=\ell+1}^{L} \mathbb{E}\big[UB_{ikt}^{(pw)}(\ell)\ x_{ikt}^{Alg^*}(\ell) - q_{ikt}(\ell)\ x_{ikt}^{Alg^*}(\ell)\big] \\
&+ \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\sum_{t=\ell+1}^{L} \mathbb{E}\big[q_{ikt}(\ell)\ x_{ikt}^{Aux^*}(\ell) - UB_{ikt}^{(pw)}(\ell)\ x_{ikt}^{Aux^*}(\ell)\big],
\end{aligned} \tag{2.21}$$

where the first equality holds by the law of total expectation. The second equality holds by $\mathbb{P}(x_i^{Alg^*}(\ell)|\bar{\mathcal{H}}_{i\ell}) = \mathbb{P}(x_i^{Aux^*}(\ell)|\bar{\mathcal{H}}_{i\ell})$ and knowing that $UB_{ikt}^{(pw)}(\ell)$ is a deterministic function given the history $\bar{\mathcal{H}}_{i\ell}$.

According to Proposition II.5, the following confidence bound holds with probability at least $1-2\delta$:

$$
\begin{aligned}
&\left|p\,\langle\,\phi_{ik^*t^*}(\ell),\ w\rangle - \langle\,\phi_{ik^*t^*}(\ell), \hat{w}_i^c(\ell)\rangle\right| \\
&\le \|\phi_{ik^*t^*}(\ell)\|_{V_{i\ell}^{-1}}\left((c_Q+\sigma_\xi)\sqrt{2\log\left(\frac{\det(V_{i\ell})^{1/2}\det(\lambda I)^{-1/2}}{\delta}\right)} + \Sigma^{(pw)}_{ik^*t^*}(\ell)\right),
\end{aligned}
$$

where $\Sigma^{(pw)}_{ik^*t^*}(\ell) = \ c_Q\Big(\sum_{s=\max\{1,\ell-\Delta\}}^{\ell-1}\sum_{j=1}^{M_s}\|\phi_{jk^*t^*}(s)\|_{V_{i\ell}^{-1}} + \big(\sum_{j=1}^{i-1}\|\phi_{jk^*t^*}(\ell)\|_{V_{i\ell}^{-1}}\big)\mathbb{1}_{i\neq 1}\Big) + \lambda^{1/2}$.

Accordingly, the two terms in (2.21) can be bounded with probability at least $1-2\delta$. For the first term in (2.21), we have:

$$
\begin{aligned}
&\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\sum_{t=\ell+1}^{L}\mathbb{E}\big[UB^{(pw)}_{ikt}(\ell)\ x_{ikt}^{Alg^*}(\ell) - q_{ikt}(\ell)\ x_{ikt}^{Alg^*}(\ell)\big] \le \\
&\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\sum_{t=\ell+1}^{L}\mathbb{E}\big[UB^{(pw)}_{ikt}(\ell)\ x_{ikt}^{Alg^*}(\ell) - LB^{(pw)}_{ikt}(\ell)\ x_{ikt}^{Alg^*}(\ell)\big].
\end{aligned}
$$

Similar to our arguments in Lemma II.2, we have:

$$
\begin{aligned}
&\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\sum_{t=\ell+1}^{L}\mathbb{E}\big[UB^{(pw)}_{ikt}(\ell)\ x_{ikt}^{Alg^*}(\ell) - LB^{(pw)}_{ikt}(\ell)\ x_{ikt}^{Alg^*}\ell)\big] \\
&\le 2\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\mathbb{E}\left[\|\phi_{ik^*t^*}(\ell)\|_{V_{i\ell}^{-1}}\left((c_Q+\sigma_\xi)\sqrt{2\log\left(\frac{\det(V_{i\ell})^{1/2}\det(\lambda I)^{-1/2}}{\delta}\right)} + \Sigma^{(pw)}_{ik^*t^*}(\ell)\right)\right],
\end{aligned}
\tag{2.22}
$$

where $\Sigma^{(pw)}_{ik^*t^*}(\ell) = \ c_Q\Big(\sum_{s=\max\{1,\ell-\Delta\}}^{\ell-1}\sum_{j=1}^{M_s}\|\phi_{jk^*t^*}(s)\|_{V_{i\ell}^{-1}} + \big(\sum_{j=1}^{i-1}\|\phi_{jk^*t^*}(\ell)\|_{V_{i\ell}^{-1}}\big)\mathbb{1}_{i\neq 1}\Big) + \lambda^{1/2}$.

Following similar steps in Lemmas II.2 and II.5, we have:

$$
\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\|\phi_{ik^*t^*}(\ell)\|_{V_{i\ell}^{-1}} \le \sqrt{N_L}\sqrt{2\ (d+A)\log\left(1+\frac{c_\phi^2 N_L}{\lambda(d+A)}\right)},
$$

and,

$$
2\log\left(\frac{\det(V_{i\ell})^{1/2}\det(\lambda I)^{-1/2}}{\delta}\right) \le \ (d+A)\log\left(1+\frac{c_\phi^2 N_\ell}{\lambda(d+A)}\right) + \log\left(\frac{1}{\delta^2}\right).
$$

Then, we can derive the following bound for the right-hand side of (2.22):

$$\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\|\phi_{ik^*t^*}(\ell)\|_{V_{i\ell}^{-1}}\left((c_Q+\sigma_\xi)\sqrt{2\log\left(\frac{\det(V_{i\ell})^{1/2}\det(\lambda I)^{-1/2}}{\delta}\right)}+\Sigma^{(pw)}_{ik^*t^*}(\ell)\right)$$

$$\leq \sqrt{N_L}\sqrt{2\,(d+A)\log\left(1+\tfrac{c_\phi^2N_L}{\lambda(d+A)}\right)}\left((c_Q+\sigma_\xi)\sqrt{(d+A)\log\left(1+\tfrac{c_\phi^2N_L}{\lambda(d+A)}\right)+\log\left(\tfrac{1}{\delta^2}\right)}+\lambda^{1/2}\right)$$

$$+\,2\,c_Q(d+A)\bar{M}(1+\Delta)\log\left(1+\frac{c_\phi^2N_L}{\lambda(d+A)}\right).$$

The second term in (2.21) can be bounded by a similar technique used in Lemma II.3 as:

$$\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\sum_{t=\ell+1}^{L}\mathbb{E}\big[q_{ikt}(\ell)\;x_{ikt}^{Aux^*}(\ell)-UB_{ikt}^{(pw)}(\ell)\;x_{ikt}^{Aux^*}(\ell)\big]\leq\;2\,c_QN_L\delta.$$

Summing the bounds derived for the two terms in (2.21) completes the proof. □

Proposition II.7 (Penalty Loss for Exceeding Servers' Availability)**.** *For any $\delta>0$, the following holds with probability at least $1-\delta$.*

$$\mathbb{E}[\textsc{PenaltyLoss-II}]\leq c_Q\sigma_\eta KL\sqrt{2N_L\log\left(\frac{2}{\delta}\right)}\,.$$

Proof. Recall that the resource consumption by a customer is stochastic in our general resource allocation problem and we discussed why we should account for the possibility of exceeding resource capacity in Proposition II.3. A similar argument also holds for our advance scheduling problem in which service time of a customer is stochastic and it is realized after a customer is assigned to a server-date. The PAS-LD algorithm enforces the capacity constraints to hold only in expectation by relying on the expected service time. Accordingly, the total realized service times of customers assigned to a server-date may exceed the availability of the server on that day. Thus, we consider a *penalty* on the amount of capacity allocated in excess of the servers' availability on different days.

We assume that if a customer does not show up on the service date, the server's availability is still decreased by the expected service time of the customer. Accordingly, the expected penalty loss can be calculated as follows:

$$\mathbb{E}[\textsc{PenaltyLoss-II}]$$

$$=c_Q\sum_{k=1}^{K}\sum_{t=2}^{L}\mathbb{E}\left[\left(\sum_{\ell=1}^{t-1}\sum_{i=1}^{M_\ell}\Big(\mathcal{SU}_i(\ell)\;\mathcal{S}_{ikt}(\ell)+\big(1-\mathcal{SU}_i(\ell)\big)s_{ikt}(\ell)\Big)\;x_{ikt}^{Alg^*}(\ell)-\mathcal{C}_{kt}\right)^{+}\right],$$

where c_Q is the maximum possible value for the match quality.

To establish an upper bound for the above term, we decompose it into two terms:

$$\sum_{k=1}^{K}\sum_{t=2}^{L}\mathbb{E}\left[\left(\sum_{\ell=1}^{t-1}\sum_{i=1}^{M_\ell}\Big(\mathcal{SU}_i(\ell)\,\mathcal{S}_{ikt}(\ell)+\big(1-\mathcal{SU}_i(\ell)\big)s_{ikt}(\ell)\Big)\,x_{ikt}^{Alg^*}(\ell)-\mathcal{C}_{kt}\right)^{+}\right]$$
$$\leq \underbrace{\sum_{k=1}^{K}\sum_{t=2}^{L}\mathbb{E}\left[\left|\sum_{\ell=1}^{t-1}\sum_{i=1}^{M_\ell}\mathcal{SU}_i(\ell)\big(\mathcal{S}_{ikt}(\ell)\,x_{ikt}^{Alg^*}(\ell)-s_{ikt}(\ell)\,x_{ikt}^{Alg^*}(\ell)\big)\right|\right]}_{\text{(I)}}$$
$$+\underbrace{\sum_{k=1}^{K}\sum_{t=2}^{L}\mathbb{E}\left[\left(\sum_{\ell=1}^{t-1}\sum_{i=1}^{M_\ell}s_{ikt}(\ell)\,x_{ikt}^{Alg^*}(\ell)-\mathcal{C}_{kt}\right)^{+}\right]}_{\text{(II)}}.$$

The rest of the proof can be done by bounding each term, separately.

Term (I): First, we have:

$$\sum_{k=1}^{K}\sum_{t=2}^{L}\mathbb{E}\left[\left|\sum_{\ell=1}^{t-1}\sum_{i=1}^{M_\ell}\mathcal{SU}_i(\ell)\big(\mathcal{S}_{ikt}(\ell)\,x_{ikt}^{Alg^*}(\ell)-s_{ikt}(\ell)\,x_{ikt}^{Alg^*}(\ell)\big)\right|\right]$$
$$=\sum_{k=1}^{K}\sum_{t=2}^{L}\mathbb{E}\left[\left|\sum_{\ell=1}^{t-1}\sum_{i=1}^{M_\ell}\mathcal{SU}_i(\ell)\big(\mathcal{S}_{ikt}(\ell)-s_{ikt}(\ell)\big)x_{ikt}^{Alg^*}(\ell)\right|\right]$$
$$\leq\ \sum_{k=1}^{K}\sum_{t=2}^{L}\mathbb{E}\left[\left|\sum_{\ell=1}^{t-1}\sum_{i=1}^{M_\ell}\mathcal{SU}_i(\ell)\big(\mathcal{S}_{ikt}(\ell)-s_{ikt}(\ell)\big)\right|\right].$$

Note that $\mathcal{SU}_i(\ell)(\mathcal{S}_{ikt}(\ell)-s_{ikt}(\ell))$ is a σ_η-sub-Gaussian random variable because $\eta_{ikt}(\ell)=\mathcal{S}_{ikt}(\ell)-s_{ikt}(\ell)$ is σ_η-sub-Gaussian and $\mathcal{SU}_i(\ell)\in\{0,1\}$. By the Azuma-Hoeffding inequality for sub-Gaussian random variables and its corollary (see Lemma II.4), the following high-probability bound holds:

$$\left|\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\mathcal{SU}_i(\ell)\big(\mathcal{S}_{ikt}(\ell)-s_{ikt}(\ell)\big)\right|\leq\ \sigma_\eta\sqrt{2N_L\log\left(\frac{2}{\delta}\right)},\quad\text{with probability at least } 1-\delta. \tag{2.23}$$

$$\sum_{k=1}^{K}\sum_{t=2}^{L}\mathbb{E}\left[\left|\sum_{\ell=1}^{t-1}\sum_{i=1}^{M_\ell}\mathcal{SU}_i(\ell)\big(\mathcal{S}_{ikt}(\ell)-s_{ikt}(\ell)\big)\right|\right]\leq\ \sigma_\eta KL\sqrt{2N_L\log\left(\frac{2}{\delta}\right)}.$$

Term (II): The algorithm guarantees that the capacity constraints hold in expectation.

Then, we have:

$$\mathcal{C}_{kt} \geq \sum_{\ell=1}^{t-1} \sum_{i=1}^{M_\ell} s_{ikt}(\ell)\, x_{ikt}^{Alg^*}(\ell), \quad \forall\, k \in \mathcal{K}, \;\; \forall\, t \in \mathcal{L}\backslash\{1\}.$$

Accordingly, we have:

$$\sum_{k=1}^{K} \sum_{t=2}^{L} \mathbb{E}\left[\left(\sum_{\ell=1}^{t-1} \sum_{i=1}^{M_\ell} s_{ikt}(\ell)\, x_{ikt}^{Alg^*}(\ell) - \mathcal{C}_{kt} \right)^+ \right] \leq 0.$$

Summing the bounds derived for Terms (I) and (II) completes the proof. □

2.7.3 Appendix C. Lemmas 1, 2, 3, and Propositions 8 and 9

Lemma II.1 (**Upper Bound on Expected Total Reward of Clairvoyant Policy**). *In the general resource allocation problem, the expected total reward of the clairvoyant policy is upper bounded by the LP-based benchmark-I* ($LP_1[\varphi^{\mathcal{X}}, w]$)*.*

Proof. Let $\mathbb{1}_{ik\ell}(\varphi^{\mathcal{X}}, w) \in \{0,1\}$ be the indicator that the clairvoyant policy assigns the i^{th} customer on day ℓ to resource k for a given sequence of customer contexts $\varphi^{\mathcal{X}}$ and model parameter w. Recall that the clairvoyant policy is *feasible*. Thus, the capacity constraints should hold for any sample path of realized match quality values $\mathcal{Q} = \{\mathcal{Q}_{ik}(\ell)\}_{\ell\in\mathcal{L}, i\in\mathcal{M}_\ell, k\in\mathcal{K}}$, resource consumption values $\mathcal{S} = \{\mathcal{S}_{ik}(\ell)\}_{\ell\in\mathcal{L}, i\in\mathcal{M}_\ell, k\in\mathcal{K}}$, and possible randomization in the clairvoyant policy. Then, we have:

$$\sum_{\ell=1}^{L} \sum_{i=1}^{M_\ell} \mathcal{S}_{ik}(\ell)\; \mathbb{1}_{ik\ell}(\varphi^{\mathcal{X}}, w) \;\; \leq \;\; \mathcal{C}_k, \quad \forall\, k \in \mathcal{K}. \tag{2.24}$$

Note that the random variable $\mathcal{S}_{ik}(\ell)$ is independent of the random variable $\mathbb{1}_{ik\ell}(\varphi^{\mathcal{X}}, w)$ as the clairvoyant policy does not observe $\mathcal{S}_{ik}(\ell)$ when assigns the i^{th} customer on day ℓ to resource k.

First, we need to show that $\mathbb{E}[\mathbb{1}_{ik\ell}(\varphi^{\mathcal{X}}, w)]$ is a feasible solution for the offline LP-based benchmark-I. By taking expectation of both sides of (2.24) over random realizations of resource consumption and match quality, and other possible randomization, we have:

$$\sum_{\ell=1}^{L} \sum_{i=1}^{M_\ell} s_{ik}(\ell)\; \mathbb{E}[\mathbb{1}_{ik\ell}(\varphi^{\mathcal{X}}, w)] \;\; \leq \;\; \mathcal{C}_k, \quad \forall\, k \in \mathcal{K},$$

where the above inequality holds since $s_{ik}(\ell) = \mathbb{E}[\mathcal{S}_{ik}(\ell)]$.

Let $x_{ik}(\ell) = \mathbb{E}[\mathbb{1}_{ik\ell}(\varphi^{\mathcal{X}}, w)]$. Then, the capacity constraints in $LP_1[\varphi^{\mathcal{X}}, w]$ hold. By definition, $x_{ik}(\ell) \geq 0$ and constraints $\sum_{k=1}^{K} x_{ik}(\ell) \leq 1$ hold as well. Next, it is easy to see that the objective function of $LP_1[\varphi^{\mathcal{X}}, w]$ is equal to the expected total reward of the clairvoyant policy. Thus, the expected total reward of the clairvoyant policy is upper bounded by the offline LP-based benchmark-I. □

Lemma II.2 (Bound on Difference Between Upper and Lower Bounds of Expected Match Quality). *Let $UB_{ik}^{(w)}(\ell)$ and $LB_{ik}^{(w)}(\ell)$ be the sequences of real-valued functions of $\mathcal{H}_{i\ell}$ and feature vector $\phi_{ik}(\ell)$:*

$$UB_{ik}^{(w)}(\ell) = \min\Big\{c_Q, \max_{w \in \Theta_{i\ell}^{(w)}} \langle\, \phi_{ik}(\ell), w\rangle\Big\}, \quad LB_{ik}^{(w)}(\ell) = \max\Big\{0, \min_{w \in \Theta_{i\ell}^{(w)}} \langle\, \phi_{ik}(\ell), w\rangle\Big\},$$

where $\Theta_{i\ell}^{(w)}$ is the confidence set that contains w with high probability.

Then, for any $\delta > 0$, the following holds with probability at least $1 - \delta$.

$$\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K} \mathbb{E}\big[UB_{ik}^{(w)}(\ell)\, x_{ik}^{Alg^*}(\ell) - LB_{ik}^{(w)}(\ell)\, x_{ik}^{Alg^*}(\ell)\big]$$
$$\leq 2\sqrt{N_L}\sqrt{2\,(d+K)\log\left(1+\tfrac{c_\phi^2 N_L}{\lambda(d+K)}\right)}\left(\sigma_\xi\sqrt{(d+K)\log\left(1+\tfrac{c_\phi^2 N_L}{\lambda(d+K)}\right)+\log\left(\tfrac{1}{\delta^2}\right)}+\lambda^{1/2}\right)$$
$$+\,4\,c_Q\,(d+K)\bar{M}(1+\Delta)\log\left(1+\frac{c_\phi^2 N_L}{\lambda(d+K)}\right).$$

Proof. Since we cannot assign a customer to more than one resource, we have:

$$\begin{aligned}
&\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K} \mathbb{E}\Big[UB_{ik}^{(w)}(\ell)\, x_{ik}^{Alg^*}(\ell) - LB_{ik}^{(w)}(\ell)\, x_{ik}^{Alg^*}(\ell)\Big] \\
&= \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K} \mathbb{E}\Big[\Big(UB_{ik}^{(w)}(\ell) - LB_{ik}^{(w)}(\ell)\Big)\, \mathbb{1}_{k(i,\ell)=k}\Big] \\
&= \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \mathbb{E}\big[UB_{ik^*}^{(w)}(\ell) - LB_{ik^*}^{(w)}(\ell)\big].
\end{aligned}$$

According to Proposition II.1, for any i, ℓ, and $\delta > 0$, the following holds with probability at least $1-\delta$:

$$\begin{aligned}
&\left|\langle\, \phi_{ik^*}(\ell), w\rangle - \langle\, \phi_{ik^*}(\ell), \hat{w}_i(\ell)\rangle\right| \\
&\leq \|\phi_{ik^*}(\ell)\|_{V_{i\ell}^{-1}} \left(\sigma_\xi \sqrt{2\log\left(\frac{\det(V_{i\ell})^{1/2}\det(\lambda I)^{-1/2}}{\delta}\right)} + \Sigma_{ik^*}^{(w)}(\ell)\right),
\end{aligned}$$

where $\Sigma_{ik^*}^{(w)}(\ell) = c_Q\Big(\sum_{s=\max\{1,\ell-\Delta\}}^{\ell-1} \sum_{j=1}^{M_s} \|\phi_{jk^*}(s)\|_{V_{i\ell}^{-1}} + \big(\sum_{j=1}^{i-1} \|\phi_{jk^*}(\ell)\|_{V_{i\ell}^{-1}} \big) \mathbb{1}_{i\neq 1}\Big) + \lambda^{1/2}$.

The above term can be further simplified by the following algebra:

$$\begin{aligned}
&2\log\left(\frac{\det(V_{i\ell})^{1/2}\det(\lambda I)^{-1/2}}{\delta}\right) \\
=&\ 2\log\left(\det(V_{i\ell})^{1/2}\right) + 2\log\left(\frac{\det(\lambda I)^{-1/2}}{\delta}\right) \\
\leq&\ (d+K)\log\left(\lambda + \frac{c_\phi^2 N_\ell}{d+K}\right) + 2\log\left(\det(\lambda I)^{-1/2}\right) + 2\log\left(\frac{1}{\delta}\right) \\
=&\ (d+K)\log\left(\lambda + \frac{c_\phi^2 N_\ell}{d+K}\right) + (d+K)\log\left(\frac{1}{\lambda}\right) + 2\log\left(\frac{1}{\delta}\right) \\
=&\ (d+K)\log\left(1 + \frac{c_\phi^2 N_\ell}{\lambda(d+K)}\right) + \log\left(\frac{1}{\delta^2}\right).
\end{aligned}$$

Note that $V_{i\ell}$ is a positive definite matrix, $\mathrm{tr}(V_{i\ell})$ is equal to the summation of its eigenvalues, and $\det(V_{i\ell})$ is equal to the product of its eigenvalues. Then, by the inequality of arithmetic and geometric means, we have:

$$\det(V_{i\ell}) = \prod_{i=1}^{d+K} \zeta_i \ \leq \Big(\frac{1}{d+K}\mathrm{tr}(V_{i\ell})\Big)^{d+K} \leq \left(\lambda + \frac{c_\phi^2 N_\ell}{d+K}\right)^{d+K},$$

where ζ_i is the i^{th} eigenvalue of the matrix $V_{i\ell}$. The second inequality holds by $\mathrm{tr}(V_{i\ell}) \leq \mathrm{tr}(\lambda I) + c_\phi^2 N_\ell$.

According to the confidence bound in Proposition II.1 and definitions of $UB_{ik}^{(w)}(\ell)$ and $LB_{ik}^{(w)}(\ell)$, the following holds with probability at least $1-\delta$:

$$\begin{aligned}
&\mathbb{E}\left[UB_{ik^*}^{(w)}(\ell) - LB_{ik^*}^{(w)}(\ell)\right] \\
&\leq 2\,\mathbb{E}\left[\|\phi_{ik^*}(\ell)\|_{V_{i\ell}^{-1}} \left(\sigma_\xi \sqrt{2\log\left(\frac{\det(V_{i\ell})^{1/2}\det(\lambda I)^{-1/2}}{\delta}\right)} + \Sigma_{ik^*}^{(w)}(\ell)\right)\right], \qquad (2.25)
\end{aligned}$$

where $\Sigma_{ik^*}^{(w)}(\ell) = c_Q\Big(\sum_{s=\max\{1,\ell-\Delta\}}^{\ell-1} \sum_{j=1}^{M_s} \|\phi_{jk^*}(s)\|_{V_{i\ell}^{-1}} + \big(\sum_{j=1}^{i-1} \|\phi_{jk^*}(\ell)\|_{V_{i\ell}^{-1}} \big) \mathbb{1}_{i\neq 1}\Big) + \lambda^{1/2}$.

Based on Lemma II.5, we have:

$$\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \|\phi_{ik^*}(\ell)\|^2_{V_{i\ell}^{-1}} \;\le\; 2\log\left(\frac{\det(V_{1\ L+1})}{\det(\lambda I)}\right),$$

where $\det(V_{1\ L+1}) \le \left(\lambda + \frac{c_\phi^2 N_L}{d+K}\right)^{d+K}$.

Using the Cauchy-Schwarz inequality and the obtained bound for $\det(V_{1\ L+1})$, we have:

$$\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \|\phi_{ik^*}(\ell)\|_{V_{i\ell}^{-1}} \le \sqrt{N_L}\sqrt{\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \|\phi_{ik^*}(\ell)\|^2_{V_{i\ell}^{-1}}} \le \sqrt{N_L}\sqrt{2\ (d+K)\log\left(1+\frac{c_\phi^2 N_L}{\lambda(d+K)}\right)}.$$

Furthermore, we have:

$$\begin{aligned}
&\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \|\phi_{ik^*}(\ell)\|_{V_{i\ell}^{-1}} \Big(\sum_{s=\max\{1,\ell-\Delta\}}^{\ell-1} \sum_{j=1}^{M_s} \|\phi_{jk^*}(s)\|_{V_{i\ell}^{-1}} + \big(\sum_{j=1}^{i-1} \|\phi_{jk^*}(\ell)\|_{V_{i\ell}^{-1}}\big)\mathbb{1}_{i\neq 1}\Big) \\
&\le \frac{1}{2}\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \sum_{s=\max\{1,\ell-\Delta\}}^{\ell-1} \sum_{j=1}^{M_s} \Big(\|\phi_{ik^*}(\ell)\|^2_{V_{i\ell}^{-1}} + \|\phi_{jk^*}(s)\|^2_{V_{i\ell}^{-1}} \Big) \\
&+ \frac{1}{2}\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \Big(\sum_{j=1}^{i-1} \big(\|\phi_{ik^*}(\ell)\|^2_{V_{i\ell}^{-1}} + \|\phi_{jk^*}(\ell)\|^2_{V_{i\ell}^{-1}} \big)\Big)\mathbb{1}_{i\neq 1} \\
&\le \frac{1}{2}\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \sum_{s=\max\{1,\ell-\Delta\}}^{\ell-1} \sum_{j=1}^{M_s} \Big(\|\phi_{ik^*}(\ell)\|^2_{V_{i\ell}^{-1}} + \|\phi_{jk^*}(s)\|^2_{V_{js}^{-1}} \Big) \\
&+ \frac{1}{2}\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \Big(\sum_{j=1}^{i-1} \big(\|\phi_{ik^*}(\ell)\|^2_{V_{i\ell}^{-1}} + \|\phi_{jk^*}(\ell)\|^2_{V_{j\ell}^{-1}} \big)\Big)\mathbb{1}_{i\neq 1} \\
&\le \frac{1}{2}\bar{M}\Delta\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \|\phi_{ik^*}(\ell)\|^2_{V_{i\ell}^{-1}} + \frac{1}{2}\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \sum_{s=\max\{1,\ell-\Delta\}}^{\ell-1} \sum_{j=1}^{M_s} \|\phi_{jk^*}(s)\|^2_{V_{js}^{-1}} \\
&+ \frac{1}{2}\bar{M}\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \|\phi_{ik^*}(\ell)\|^2_{V_{i\ell}^{-1}} + \frac{1}{2}\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \Big(\sum_{j=1}^{i-1} \|\phi_{jk^*}(\ell)\|^2_{V_{j\ell}^{-1}} \Big)\mathbb{1}_{i\neq 1} \\
&\le \bar{M}\Delta\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \|\phi_{ik^*}(\ell)\|^2_{V_{i\ell}^{-1}} + \bar{M}\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \|\phi_{ik^*}(\ell)\|^2_{V_{i\ell}^{-1}},
\end{aligned}$$

where $\bar{M}$ is an upper bound on the maximum number of arrivals on each day. The first inequality holds by using $ab \le (a^2+b^2)/2$ for $a, b \in \mathbb{R}$. The second inequality holds because $\|\phi_{jk}(s)\|_{V_{i\ell}^{-1}} \le \|\phi_{jk}(s)\|_{V_{js}^{-1}}$ for all $j \le i,\ s \le \ell$. The last inequality holds because $\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{s=\max\{1,\ell-\Delta\}}^{\ell-1}\sum_{j=1}^{M_s} \|\phi_{jk^*}(s)\|^2_{V_{js}^{-1}} \le \bar{M}\Delta\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \|\phi_{ik^*}(\ell)\|^2_{V_{i\ell}^{-1}}$ and

$\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\big(\sum_{j=1}^{i-1}\|\phi_{jk^*}(\ell)\|^2_{V_{j\ell}^{-1}}\big)\mathbb{1}_{i\neq 1} \leq \bar{M}\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\|\phi_{ik^*}(\ell)\|^2_{V_{i\ell}^{-1}}$.

Now, we are ready to derive the following bound:

$$\begin{aligned}
&\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\|\phi_{ik^*}(\ell)\|_{V_{i\ell}^{-1}}\left(\sigma_\xi\sqrt{2\log\left(\frac{\det(V_{i\ell})^{1/2}\det(\lambda I)^{-1/2}}{\delta}\right)}+\Sigma_{ik^*}^{(w)}(\ell)\right)\\
&\leq \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\|\phi_{ik^*}(\ell)\|_{V_{i\ell}^{-1}}\left(\sigma_\xi\sqrt{2\log\left(\frac{\det(V_{i\ell})^{1/2}\det(\lambda I)^{-1/2}}{\delta}\right)}+\lambda^{1/2}\right)\\
&+c_Q\bar{M}(1+\Delta)\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\|\phi_{ik^*}(\ell)\|^2_{V_{i\ell}^{-1}}\\
&\leq \sqrt{N_L}\sqrt{2\,(d+K)\log\left(1+\frac{c_\phi^2 N_L}{\lambda(d+K)}\right)}\left(\sigma_\xi\sqrt{(d+K)\log\left(1+\frac{c_\phi^2 N_L}{\lambda(d+K)}\right)+\log\left(\frac{1}{\delta^2}\right)}+\lambda^{1/2}\right)\\
&\qquad+2\,c_Q\,(d+K)\bar{M}(1+\Delta)\log\left(1+\frac{c_\phi^2 N_L}{\lambda(d+K)}\right).
\end{aligned}$$

Replacing the above result into the inequality (2.25) completes the proof. □

Lemma II.3 (**Bound on Difference Between Expected Match Quality and its Upper/Lower Bound**). *For any $\delta > 0$, the following bounds hold with probability at least $1-\delta$.*

$$\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\mathbb{E}\big[q_{ik}(\ell)\,x_{ik}^{Aux^*}(\ell)-UB_{ik}^{(w)}(\ell)\,x_{ik}^{Aux^*}(\ell)\big]\;\leq c_Q N_L\delta.$$

and,

$$\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\mathbb{E}\big[LB_{ik}^{(w)}(\ell)\,x_{ik}^{Aux^*}(\ell)-q_{ik}(\ell)\,x_{ik}^{Aux^*}(\ell)\big]\;\leq c_Q N_L\delta.$$

Proof. Since we cannot assign each customer to more than one resource, we have

$$\begin{aligned}
&\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\mathbb{E}\left[q_{ik}(\ell)\,x_{ik}^{Aux^*}(\ell)-UB_{ik}^{(w)}(\ell)\,x_{ik}^{Aux^*}(\ell)\right]\\
&=\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\mathbb{E}\Big[\Big(q_{ik}(\ell)-UB_{ik}^{(w)}(\ell)\Big)\mathbb{1}_{k(i,\ell)=k}\Big]\\
&=\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\mathbb{E}\big[q_{ik^*}(\ell)-UB_{ik^*}^{(w)}(\ell)\big].
\end{aligned}$$

Since $q_{ik}(\ell) \leq c_Q$, we have:

$$\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\big(q_{ik^*}(\ell) - UB_{ik^*}^{(w)}(\ell)\big) \leq c_Q \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\mathbb{1}\big(q_{ik^*}(\ell) > UB_{ik^*}^{(w)}(\ell)\big).$$

Taking expectation from both sides results in the following:

$$\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\mathbb{E}\big[q_{ik^*}(\ell) - UB_{ik^*}^{(w)}(\ell)\big] \leq c_Q \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\mathbb{P}\big(q_{ik^*}(\ell) \; > \; UB_{ik^*}^{(w)}(\ell)\big).$$

Next, we define the sequences of $\overline{UB}_{ik}^{(w)}(\ell)$ and $\overline{LB}_{ik}^{(w)}(\ell)$ as follows:

$$\overline{UB}_{ik}^{(w)}(\ell) = \max_{w \in \Theta_{i\ell}^{(w)}} \langle\, \phi_{ik}(\ell), w\rangle, \quad \overline{LB}_{ik}^{(w)}(\ell) = \min_{w \in \Theta_{i\ell}^{(w)}} \langle\, \phi_{ik}(\ell), w\rangle.$$

Recall from Proposition II.1 that for any i, ℓ, and $\delta > 0$, with probability at least $1-\delta$, we have:

$$\big|\langle\, \phi_{ik^*}(\ell), w\rangle - \langle\, \phi_{ik^*}(\ell), \hat{w}_i(\ell)\rangle\big| \leq \|\phi_{ik^*}(\ell)\|_{V_{i\ell}^{-1}} \left(\sigma_\xi \sqrt{2\log\left(\frac{\det(V_{i\ell})^{1/2}\det(\lambda I)^{-1/2}}{\delta}\right)} + \Sigma_{ik^*}^{(w)}(\ell)\right).$$

The above statement is equivalent to the following:

$$\mathbb{P}\left[\overline{LB}_{ik^*}^{(w)}(\ell) \leq q_{ik^*}(\ell) \leq \overline{UB}_{ik^*}^{(w)}(\ell)\right] \; \geq 1-\delta.$$

Note that if $UB_{ik^*}^{(w)}(\ell) < c_Q$, then $UB_{ik^*}^{(w)}(\ell) = \overline{UB}_{ik^*}^{(w)}(\ell)$ by definition of $UB_{ik^*}^{(w)}(\ell)$. When $q_{ik^*}(\ell) > UB_{ik^*}^{(w)}(\ell)$, we have $UB_{ik^*}^{(w)}(\ell) < c_Q$, which implies that $UB_{ik^*}^{(w)}(\ell) = \overline{UB}_{ik^*}^{(w)}(\ell)$. Accordingly, the proof of the first statement of the lemma is completed by the following bound:

$$\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\mathbb{P}\big(q_{ik^*}(\ell) \; > \; UB_{ik^*}^{(w)}(\ell)\big) \; = \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\mathbb{P}\big(q_{ik^*}(\ell) \; > \; \overline{UB}_{ik^*}^{(w)}(\ell)\big) \; \leq N_L\delta.$$

Similarly, the following bound can be established for the second statement of the lemma:

$$\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K}\mathbb{E}\big[LB_{ik}^{(w)}(\ell)\; x_{ik}^{Aux^*}(\ell) - q_{ik}(\ell)\; x_{ik}^{Aux^*}(\ell)\big] \; \leq c_Q N_L \delta.$$

□

Proposition II.8 (**Competitive Ratio of Mechanism 1**). *Let V^{AUX_1} be the total expected reward obtained by Mechanism 1 for solving the auxiliary problem, and let V^{BM_1} be the total expected reward of the LP-based benchmark-I. Then, the following holds:*

$$\frac{V^{AUX_1}}{V^{BM_1}} \geq \frac{1-\eta_{\max}\Lambda}{1+\beta} \,,$$

where $\eta_{\max} = \max\limits_{i,k,\ell}\{\frac{r_{ik}(\ell)}{s_{ik}(\ell)}\}$, $\Lambda = \max\limits_{i,k,\ell}(\frac{s_{ik}(\ell)}{\mathcal{C}_k})$, $\Gamma \;=\; (1+\Lambda)^{1/\Lambda}$, *and* $\beta = \frac{\eta_{\max}}{\Gamma-1}$.

Remark. *When $\Lambda \to 0$ and $\eta_{\max} \to 1$, then coefficient $\beta \to 1/(e-1)$. Thus, the above ratio converges to $1-1/e$, which recovers the classical result in the primal-dual paradigm.*

Proof. The aim is to derive the competitive ratio of Mechanism 1 for solving the auxiliary problem, in which the model parameter w is known and there is no need for learning. Our analysis is based on the primal-dual paradigm which maintains a set of dual variables to guide the primal solutions.

We formulate primal and dual problems in which the model parameter w and the sequence of customer contexts $\varphi^{\mathcal{X}}$ are known in advance.

Primal Problem:

$$\begin{aligned}
\max_{x} \quad & \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\sum_{k=1}^{K} r_{ik}(\ell)\, x_{ik}(\ell) && \\
\text{s.t.} \quad & \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} s_{ik}(\ell)\, x_{ik}(\ell) \;\leq\; \mathcal{C}_k, \quad \forall\, k \in \mathcal{K} && (2.26)\\
& \sum_{k=1}^{K} x_{ik}(\ell) \;\leq\; 1, \quad \forall\, \ell \in \mathcal{L},\; \forall\, i \in \mathcal{M}_\ell && (2.27)\\
& x_{ik}(\ell) \;\geq\; 0, \quad \forall\, \ell \in \mathcal{L},\; \forall\, i \in \mathcal{M}_\ell,\; \forall\, k \in \mathcal{K}, &&
\end{aligned}$$

where $x_{ik}(\ell)$ is corresponding to the probability of assigning the i^{th} customer on day ℓ to resource k.

We construct the dual problem by defining dual variables y_k and $\theta_i(\ell)$ corresponding to constraints (2.26) and (2.27), respectively.

Dual Problem:

$$\begin{aligned}
\min_{y,\theta} \quad & \sum_{k=1}^{K} y_k\, \mathcal{C}_k + \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \theta_i(\ell) && \\
\text{s.t.} \quad & s_{ik}(\ell)\, y_k + \theta_i(\ell) \;\geq\; r_{ik}(\ell), \quad \forall\, \ell \in \mathcal{L},\; \forall\, i \in \mathcal{M}_\ell,\; \forall\, k \in \mathcal{K} && (2.28)\\
& y_k,\; \theta_i(\ell) \;\geq\; 0, \quad \forall\, \ell \in \mathcal{L},\; \forall\, i \in \mathcal{M}_\ell,\; \forall\, k \in \mathcal{K}. &&
\end{aligned}$$

The proof consists of two main steps. In Step 1, at each iteration (per arrival of a customer), we show that the dual solution is always *feasible*, the primal solution is *almost feasible*, and the ratio between the change in the objective value of the dual solution and the change in the objective value of the primal solution is less than $1+\beta$. In Step 2, we construct a feasible primal solution for the resource allocation mechanism and bound its competitive ratio.

Step 1 (Feasibility and Primal-Dual Change Ratio). The idea of the proof to show the feasibility of the dual solution is similar to the proof of Theorem 3.1 in [25] for the Adwords problem. However, we need to adapt it for our general resource allocation problem in which customer rewards are not necessarily the same as the resource consumption values.

Part I: First, we prove that the resource allocation mechanism provides a feasible dual solution.

At each iteration, the resource allocation mechanism sets the dual variable $\theta_i(\ell)$ to $r_{ik^*}(\ell) - s_{ik^*}(\ell)\ y_{k^*}$. Due to the acceptance criterion (i.e., $r_{ik^*}(\ell) - s_{ik^*}(\ell)\ y_{k^*} \geq 0$) used in the mechanism, it is easy to see that constraint (2.28) holds and $\theta_i(\ell) \geq\ 0$. Note that $y_k \geq 0$ due to the multiplicative updating equation for this variable. Hence, the mechanism provides a feasible dual solution at each iteration.

Part II: Next, we prove that the mechanism provides an *almost* feasible primal solution at each iteration.

First, note that constraint (2.27) holds since a customer must be either rejected or assigned to only one resource. Also, $x_{ik}(\ell) \geq 0$ because its value is either 1 or 0. It remains to show constraint (2.26) holds.

Let $\mathcal{I}_k$ be the set containing (i, ℓ) pairs corresponding to the indices of the customers assigned to resource k. Let $\mathcal{J} = \{1, \cdots, J^{\text{end}}\}$ be the set of iterations, and J_k be the last iteration that y_k is updated using the multiplicative equation. For notational convenience, we write J_k as J and then we have $y_k^{(J)}$ as the last value of y_k. We also define (i_j, ℓ_j) pair to refer to the customer at iteration j. We need to show that when $\sum_{(i,\ell)\in\mathcal{I}_k} s_{ik}(\ell)\ x_{ik}(\ell) \geq \mathcal{C}_k$ and $x_{ik}(\ell) = 1$ for $(i, \ell) \in \mathcal{I}_k$, an arriving customer cannot be assigned to resource k anymore. Then, it suffices to show that the following holds for any $\ell \in \mathcal{L}$, $i \in \mathcal{M}_\ell$, and $k \in \mathcal{K}$:

$$\text{If} \sum_{(n,s)\in\mathcal{I}_k} s_{nk}(s)\ x_{nk}(s)\ \geq\ \mathcal{C}_k, \ \text{ then } \ \left(r_{ik}(\ell) - s_{ik}(\ell)\ y_k^{(J)}\right) <\ 0. \tag{2.29}$$

Note that there can be at most one iteration in which the above condition is violated. This happens when there is available capacity before assigning a customer but it is less than the expected resource consumption value of the customer. We carefully take care of this possibility in Step 2.

In the resource allocation mechanism, we have the following multiplicative updating equation for dual variable y_k at any relevant iteration j:

$$y_k^{(j)} \;=\; y_k^{(j-1)}\Big(1+\frac{s_{i_jk}(\ell_j)}{\mathcal{C}_k}\Big) \;+\; \beta\;\Big(\frac{r_{i_jk}(\ell_j)}{\mathcal{C}_k}\Big).$$

Let $\eta_{min} \;=\; \min\limits_{i,k,\ell}\{\frac{r_{ik}(\ell)}{s_{ik}(\ell)}\}$. Since we can uniformly scale the units of $r_{ik}(\ell)$ by constant $1/\eta_{min}$, without loss of generality, we assume $\frac{r_{ik}(\ell)}{s_{ik}(\ell)} \geq 1$ for each $\ell \in \mathcal{L}$, $i \in \mathcal{M}_\ell$, $\;k \in \mathcal{K}$. Then, we have:

$$y_k^{(j)}+\beta \;\geq\; (y_k^{(j-1)}+\beta)\Big(1+\frac{s_{i_jk}(\ell_j)}{\mathcal{C}_k}\Big). \tag{2.30}$$

We define $\Lambda = \max\limits_{i,k,\ell}\big(\frac{s_{ik}(\ell)}{\mathcal{C}_k}\big)$ and $\Gamma \;=\; (1+\Lambda)^{1/\Lambda}$. Then, we have:

$$1+\frac{s_{ik}(\ell)}{\mathcal{C}_k} \geq \;\Gamma^{\Big(\frac{s_{ik}(\ell)}{\mathcal{C}_k}\Big)}, \quad \forall\, \ell\in\mathcal{L},\;\; i\in\mathcal{M}_\ell,\;\; k\in\mathcal{K}, \tag{2.31}$$

where the inequality holds by using $\frac{1}{m}\ln(1+m) \geq \frac{1}{n}\ln(1+n)$ for any $0 \;\leq m \;\leq\; n \;\leq\; 1$, and having $m = \dfrac{s_{ik}(\ell)}{\mathcal{C}_k}$ and $n = \Lambda$. Note that the assumption of $0 \;\leq m \;\leq\; n \;\leq\; 1$ holds because clearly $0 \leq m \leq n$ and the expected resource consumption value of a customer is by far less than the total capacity of a resource ($n \leq 1$).

Plugging (2.31) into (2.30) and using a recursion technique yield the following:

$$y_k^{(J)}+\beta \;\geq\; (y_k^{(0)}+\beta)\;\Gamma^{\sum_{(i,\ell)\in\mathcal{I}_k}\Big(\frac{s_{ik}(\ell)}{\mathcal{C}_k}\Big)} \;=\; \beta\,\Gamma^{\sum_{(i,\ell)\in\mathcal{I}_k}\Big(\frac{s_{ik}(\ell)}{\mathcal{C}_k}\Big)},$$

where the equality holds because the initial value of y_k is zero.

Next, the following holds because we have $\sum_{(i,\ell)\in\mathcal{I}_k} s_{ik}(\ell)\;x_{ik}(\ell) \geq \mathcal{C}_k$, and $x_{ik}(\ell) = 1$ for $(i,\ell) \in \mathcal{I}_k,\; k \in \mathcal{K}$:

$$y_k^{(J)} \;\geq \beta\;(\Gamma-1).$$

Recall that the resource allocation mechanism is required to meet (2.29), which implies that when the capacity constraint is exceeded, the acceptance criterion does not hold. To make sure that this condition holds, it suffices to have:

$$\beta\;(\Gamma-1) \;\geq\; \frac{r_{ik}(\ell)}{s_{ik}(\ell)}\;, \quad \forall\;\; \ell\in\mathcal{L},\;\; i\in\mathcal{M}_\ell,\;\; \forall\;\; k\in\mathcal{K}.$$

The appropriate choice for β is $\frac{\eta_{\max}}{(\Gamma-1)}$, where $\eta_{\max} = \max\limits_{i,k,\ell}\{\frac{r_{ik}(\ell)}{s_{ik}(\ell)}\}$.

<u>Part III</u>: Lastly, we prove that the ratio between the change in the dual objective function and the change in the primal objective function is less than $1 + \beta$ at each iteration.

Recall that (i_j, ℓ_j) pair refers to the customer assigned to resource k^* at iteration j. Also, let Obj_P and Obj_D be the objective values of the primal and dual solutions, respectively. Then, the change in the objective value of the primal solution at the j^{th} iteration $\Delta(Obj)_P^{(j)}$ is $r_{i_j k^*}(\ell_j)$. Similarly, the change in the objective value of the dual solution at the j^{th} iteration $\Delta(Obj)_D^{(j)}$ can be calculated as follows:

$$\begin{aligned}
\Delta(Obj)_D^{(j)} &= \mathcal{C}_{k^*}\ \Delta(y_{k^*}^{(j)}) + \theta_{i_j}(\ell_j) \\
&= \mathcal{C}_{k^*}\left(y_{k^*}^{(j-1)}\Big(\frac{s_{i_j k^*}(\ell_j)}{\mathcal{C}_{k^*}}\Big)\ +\ \beta\ \Big(\frac{r_{i_j k^*}(\ell_j)}{\mathcal{C}_{k^*}}\Big)\right) + \Big(r_{i_j k^*}(\ell_j) - s_{i_j k^*}(\ell_j)\ y_{k^*}^{(j-1)}\Big) \\
&= r_{i_j k^*}(\ell_j)\ (1+\beta).
\end{aligned}$$

Since $Obj_P = \sum_{j\in\mathcal{J}} \Delta(Obj)_P^{(j)}$ and $Obj_D = \sum_{j\in\mathcal{J}} \Delta(Obj)_D^{(j)}$, we have $Obj_D/Obj_P \leq 1+\beta$.

Step 2 (Competitive Ratio). At first, we construct a feasible primal solution for the resource allocation mechanism. Then, we bound the ratio between the objective value of the feasible primal solution and the objective value of the almost feasible primal solution. Finally, we bound the competitive ratio.

We start by constructing a feasible primal solution for the resource allocation mechanism. For each $k \in \mathcal{K}$, we define $x_k = \{x_{ik}(\ell)\}_{(i,\ell)\in\mathcal{I}_k}$ as the *almost* feasible primal solution obtained by the mechanism, and $\tilde{x}_k$ as the feasible primal solution obtained by tweaking x_k. It should be noted that we have $\tilde{x}_{ik}(\ell) = x_{ik}(\ell)$ for $(i,\ell) \in \mathcal{I}_k \backslash \{(i_J, \ell_J)\}$ and $\tilde{x}_{i_J k}(\ell_J) \leq x_{i_J k}(\ell_J)$. Let $\widetilde{Obj}_P$ be the objective value of the feasible primal solution obtained by converting the almost feasible solution of the mechanism, and $Obj_{\,BM_1}$ be the objective value of the optimal solution of the offline LP-based benchmark-I. By weak duality, we have $Obj_{\,BM_1} \leq Obj_D$. Then, the following holds:

$$\frac{\widetilde{Obj}_P}{Obj_{\,BM_1}} \geq \frac{\widetilde{Obj}_P}{Obj_D} = \frac{\widetilde{Obj}_P\ Obj_P}{Obj_P\ Obj_D} = \frac{\widetilde{Obj}_P/Obj_P}{Obj_D/Obj_P}. \tag{2.32}$$

By Part III in Step 1, we know that $\frac{Obj_D}{Obj_P} \leq (1+\beta)$. Thus, we need to find a tight lower bound for $\frac{\widetilde{Obj}_P}{Obj_P}$. In the following, we compute the ratio between the objective value of the feasible primal solution $\tilde{x}$ and the objective value of the almost feasible primal solution x for each resource k.

$$\begin{aligned}
\frac{\widetilde{Obj}_P^{(k)}}{Obj_P^{(k)}} = \frac{\sum_{(i,\ell)\in\mathcal{I}_k} r_{ik}(\ell)\ \tilde{x}_{ik}(\ell)}{\sum_{(i,\ell)\in\mathcal{I}_k} r_{ik}(\ell)\ x_{ik}(\ell)} &= \frac{\sum_{(i,\ell)\in\mathcal{I}_k} r_{ik}(\ell)\ x_{ik}(\ell) - r_{i_Jk}(\ell_J)\ x_{i_Jk}(\ell_J) + r_{i_Jk}(\ell_J)\ \tilde{x}_{i_Jk}(\ell_J)}{\sum_{(i,\ell)\in\mathcal{I}_k} r_{ik}(\ell)\ x_{ik}(\ell)} \\
&= 1 - \frac{r_{i_Jk}(\ell_J)\big(x_{i_Jk}(\ell_J) - \tilde{x}_{i_Jk}(\ell_J)\big)}{\sum_{(i,\ell)\in\mathcal{I}_k} r_{ik}(\ell)\ x_{ik}(\ell)} \\
&= 1 - \frac{\Big(\frac{r_{i_Jk}(\ell_J)}{s_{i_Jk}(\ell_J)}\Big) s_{i_Jk}(\ell_J)\big(x_{i_Jk}(\ell_J) - \tilde{x}_{i_Jk}(\ell_J)\big)}{\sum_{(i,\ell)\in\mathcal{I}_k} \Big(\frac{r_{ik}(\ell)}{s_{ik}(\ell)}\Big) s_{ik}(\ell)\ x_{ik}(\ell)} \\
&\geq 1 - \Big(\frac{\eta_{\max}}{\eta_{min}}\Big) \frac{s_{i_Jk}(\ell_J)\big(x_{i_Jk}(\ell_J) - \tilde{x}_{i_Jk}(\ell_J)\big)}{\sum_{(i,\ell)\in\mathcal{I}_k} s_{ik}(\ell)\ x_{ik}(\ell)},
\end{aligned}$$

where $\eta_{min} = \min\limits_{i,k,\ell}\{\frac{r_{ik}(\ell)}{s_{ik}(\ell)}\}$ and $\eta_{\max} = \max\limits_{i,k,\ell}\{\frac{r_{ik}(\ell)}{s_{ik}(\ell)}\}$.

Note that $x_{i_Jk}(\ell_J) - \tilde{x}_{i_Jk}(\ell_J)$ is either 1 or 0. Thus, we have:

$$\frac{\widetilde{Obj}_P^{(k)}}{Obj_P^{(k)}} \geq 1 - \Big(\frac{\eta_{\max}}{\eta_{min}}\Big) \frac{s_{i_Jk}(\ell_J)}{\sum_{(i,\ell)\in\mathcal{I}_k} s_{ik}(\ell)\ x_{ik}(\ell)}.$$

Accordingly, we have:

$$\frac{\widetilde{Obj}_P}{Obj_P} \geq \min_{k\in\mathcal{K}} \left(\frac{\widetilde{Obj}_P^{(k)}}{Obj_P^{(k)}}\right) \geq \min_{k\in\mathcal{K}} \left(1 - \Big(\frac{\eta_{\max}}{\eta_{min}}\Big) \frac{s_{i_Jk}(\ell_J)}{\sum_{(i,\ell)\in\mathcal{I}_k} s_{ik}(\ell)\ x_{ik}(\ell)}\right) = 1 - \eta_{\max}\Lambda\ ,$$

where the second equality holds by $\sum_{(i,\ell)\in\mathcal{I}_k} s_{ik}(\ell)\ x_{ik}(\ell) \geq \mathcal{C}_k$ and $\frac{s_{i_Jk}(\ell_J)}{\sum_{(i,\ell)\in\mathcal{I}_k} s_{ik}(\ell)\ x_{ik}(\ell)} \leq \frac{s_{i_Jk}(\ell_J)}{\mathcal{C}_k} \leq \max\limits_{i,k,\ell}\big(\frac{s_{ik}(\ell)}{\mathcal{C}_k}\big) = \Lambda$.

The proof is completed by the following bound:

$$\frac{\widetilde{Obj}_P}{Obj_{\ BM_1}} \geq \frac{\widetilde{Obj}_P/Obj_P}{Obj_D/Obj_P} \geq \frac{1-\eta_{\max}\Lambda}{1+\beta},$$

where the first inequality holds by (2.32). □

Proposition II.9 (Competitive Ratio of Mechanism 2). *Let V^{AUX_2} be the total expected reward obtained by Mechanism 2 for solving the auxiliary problem, and let V^{BM_2} be the total expected reward of the LP-based benchmark-II. Then, the following holds:*

$$\frac{V^{AUX_2}}{V^{BM_2}} \geq \frac{1-\widetilde{\eta}_{\max}\widetilde{\Lambda}}{1+\widetilde{\beta}}\ ,$$

where $\widetilde{\eta}_{\max} = \max\limits_{i,k,t,\ell}\{\frac{r_{ikt}(\ell)}{s_{ikt}(\ell)}\}$, $\widetilde{\Lambda} = \max\limits_{i,k,t,\ell}(\frac{s_{ikt}(\ell)}{\mathcal{C}_{kt}})$, $\widetilde{\Gamma} \ = \ (1+\widetilde{\Lambda})^{1/\widetilde{\Lambda}}$, *and* $\widetilde{\beta} = \frac{\widetilde{\eta}_{\max}}{\widetilde{\Gamma}-1}$.

Remark. *When* $\widetilde{\Lambda} \to 0$ *and* $\widetilde{\eta}_{\max} \to 1$, *then coefficient* $\widetilde{\beta} \to 1/(\mathrm{e}-1)$. *Thus, the above ratio converges to* $1 - 1/e$.

Proof. The proof is similar to the proof of Proposition II.8 and we omit the details here.

2.7.4 Appendix D. Known Results

In this section, we provide some known results from the literature. For completeness, we provide self-contained and more expository version of the original proof in Lemma II.5.

Lemma II.4 (**Azuma-Hoeffding for Sub-Gaussian Random Variables**)**.** *Let* $Y_1, \cdots, Y_n$ *be a* σ*-sub-Gaussian martingale difference sequence adapted to* $X_1, \cdots, X_n$ *such that:*

$$\mathbb{E}\Big[\exp\,(\lambda Y_i) \mid X_{i-1}\Big] \leq \exp\Big(\frac{\lambda^2\sigma^2}{2}\Big), \quad \textit{for all } \lambda \in \mathbb{R}.$$

Then, for every $t > 0$, *we have:*

$$\mathbb{P}\Big(\big|\sum_{i=1}^{n} Y_i\big| \geq t\Big) \leq 2\,\exp\Big(-\frac{t^2}{2n\sigma^2}\Big).$$

As a corollary of the Azuma-Hoeffding inequality, we have the following bound:

$$\big|\sum_{i=1}^{n} Y_i\big| \leq \sigma\sqrt{2n\log\left(\frac{2}{\delta}\right)}\,, \quad \text{with probability at least } 1-\delta.$$

Lemma II.5 (**Upper Bound on Summation of Feature Vectors**)**.** *Let* $\{\phi_{ik}(\ell)\}_{\ell\in\mathcal{L}, i\in\mathcal{M}_\ell}$ *be a sequence of feature vectors in* $\mathbb{R}^{d+K}$. *When* $\lambda_{\min}(V_{i\ell})$ *is large enough (i.e.,* $\lambda_{\min}(V_{i\ell}) \geq \max\{1, c_\phi^2\}$*), the following holds almost surely (Adopted from Lemma 9 in [41]):*

$$\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell} \|\phi_{ik}(\ell)\|^2_{V_{i\ell}^{-1}} \leq 2\log\left(\frac{\det(V_{1\ L+1})}{\det(\lambda I)}\right).$$

Proof. The determinant of $V_{1\ L+1}$ can be calculated by the following iterative technique:

$$\begin{aligned}
\det(V_{1\ L+1}) &= \det(V_{M_L L} + \phi_{M_L k}(L)\ \phi^{'}_{M_L k}(L)) \\
&= \det\left(V^{1/2}_{M_L L}\left(I + V^{-1/2}_{M_L L}\ \phi_{M_L}(L)\ \phi^{'}_{M_L k}(L)\ V^{-1/2}_{M_L L}\right)V^{1/2}_{M_L L}\right) \\
&= \det(V_{M_L L}) \det\left(I + \left(V^{-1/2}_{M_L L}\ \phi_{M_L k}(L)\right)\left(V^{-1/2}_{M_L L}\ \phi_{M_L k}(L)\right)^{'}\right) \\
&= \det(V_{M_L L})\left(1 + \|\phi_{M_L k}(L)\|^2_{V^{-1}_{M_L L}}\right) = \det(\lambda I)\left[\prod_{\ell=1}^{L}\prod_{i=1}^{M_\ell}\left(1 + \|\phi_{ik}(\ell)\|^2_{V^{-1}_{i\ell}}\right)\right],
\end{aligned} \tag{2.33}$$

where the fourth equality holds because all the eigenvalues of a matrix of the form $I + xx^{'}$ where $x \in \mathbb{R}^n$ are one except the one which is $1 + \|x\|^2$. The last equality is obtained by recursion.

Taking the logarithm of (2.33) results in the following:

$$\log\big(\det(V_{1\ L+1})\big) = \log\big(\det(\lambda I)\big) + \sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\log\left(1 + \|\phi_{ik}(\ell)\|^2_{V^{-1}_{i\ell}}\right).$$

Since $x \ \leq\ 2\ \log(1+x)$ for $0 \leq x \leq 1$, we have:

$$\begin{aligned}
\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\min\left\{1, \|\phi_{ik}(\ell)\|^2_{V^{-1}_{i\ell}}\right\} &\leq 2\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\log\left(1 + \min\left\{1, \|\phi_{ik}(\ell)\|^2_{V^{-1}_{i\ell}}\right\}\right) \\
&\leq 2\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\log\left(1 + \|\phi_{ik}(\ell)\|^2_{V^{-1}_{i\ell}}\right) = 2\log\left(\frac{\det(V_{1\ L+1})}{\det(\lambda I)}\right).
\end{aligned}$$

Note that we have $\|\phi_{ik}(\ell)\|^2_{V^{-1}_{i\ell}} \leq \lambda^{-1}_{\min}(V_{i\ell})\ \|\phi_{ik}(\ell)\|^2 \leq \lambda^{-1}_{\min}(V_{i\ell})\ c^2_\phi$. Then, $\|\phi_{ik}(\ell)\|^2_{V^{-1}_{i\ell}} \leq 1$ holds when $\lambda_{\min}(V_{i\ell})$ is large enough (i.e., $\lambda_{\min}(V_{i\ell}) \geq \max\{1, c^2_\phi\}$). Finally, we have:

$$\sum_{\ell=1}^{L}\sum_{i=1}^{M_\ell}\|\phi_{ik}(\ell)\|^2_{V^{-1}_{i\ell}} \leq 2\log\left(\frac{\det(V_{1\ L+1})}{\det(\lambda I)}\right).$$

□

CHAPTER III

Personalized Hospital Admission Control: A Contextual Learning Approach [1]

3.1 Introduction

The choice of care unit upon admission to the hospital matters and is complicated by the limited unit capacity, high variability in patients' health status, and the high utilization of intensive and intermediate care units. The intensive care unit (ICU) is the most expensive care unit, consuming 15%-40% of hospital costs ([57], [93]). An ICU provides specialized care for critical patients and has the highest ratio of patients to nurses. The step-down unit (SDU), also known as intermediate care unit, is a less expensive unit which provides a lower level of staffing and care. Generally, patients who can be treated in the SDU can be treated in the ICU as well, while patients in the SDU receive a lower level of care compared to the patients in the ICU ([28]). Although there are useful guidelines for critical care admissions based on patient's physical conditions and requirements, there is still a large grey area regarding the admission decisions ([92]).

There are some studies in the literature investigating the various patient health outcomes (e.g., mortality risk and readmission risk) associated with different levels of care (see e.g., [69] and [28]). Unplanned hospital readmissions have drawn substantial attention over the past decade, because they reflect poor health outcomes and are deemed to be unnecessarily high and costly. According to the Medicare payment advisory commission, almost 20% of Medicare discharges are readmitted within 30 days, accounting for 15 to 20 billion dollars ([52]). Surprisingly, it has been shown in several studies that up to 75% of all readmissions have been judged to be preventable ([17]). Many hospitals are eager to improve and reduce their readmission rate for a variety of reasons. These include bad publicity, unreimbursed

[1] Mohammad Zhalechian, Esmaeil Keyvanshokooh, Cong Shi, and Mark P Van Oyen. Personalized hospital admission control: a contextual learning approach. *Operations Research*, 2022.

expenses (e.g., for procedures in a bundled payments agreement), and in some cases the threat of penalties for hospitals with excessive readmission rates under the Affordable Care Act (ACA). Using econometric approaches, it has been shown that there is a relationship between readmissions and the care unit placement decisions, and the benefits associated with different levels of care can be highly heterogeneous due to the different needs of patients ([28]). The literature has emphasized stochastic models and queueing analysis and control to provide various models of congestion in the care units (see e.g., [69] and [60]). Our scope is focused on providing an approach based on learning with optimization as an unexplored direction that incorporates recent development in the field. In this chapter, we focus on hospital readmission as a way to identify the limits to which an online algorithm for care unit placement can improve the readmission rate; however, our methodology can be adapted to other patient outcomes (e.g., mortality risk) as well.

In current practice, care unit placement (admission decision) highly depends on the training and the experience of the physicians and staff ([87], [32]). The patient-specific nature of admission decisions and the wide variety of patient characteristics make it almost impossible for a physician to have a reliable estimation of the patient's health outcome based on the experience and the available data for the patient. On the other hand, there is a fundamental trade-off between the benefit of assigning patients to an ICU or SDU and the loss of an available bed for a more deserving patient arriving in the near future. As a response to these challenges, the health community has raised the need to develop new strategies for care unit placement decisions ([65], [31]).

Traditional readmission risk prediction and admission control employs offline estimation of patient severity, where all the input-output data should be collected first and then model parameters can be estimated. Both offline and online methods rely on historical data to provide estimates; however, online methods offer the advantage of rapid adaptive learning. The key in this type of learning is adaptive data collection through striking a trade-off between taking advantage of our current beliefs to make decisions (exploitation) and learning more about poorly estimated actions (exploration).

We address the fundamental question of how to pursue the goal of readmission reduction through admission control when there is uncertainty regarding the needs of patients. To investigate this question, we answer two follow-up research questions: First, how can one develop a personalized admission control system that can (i) adaptively learn readmission risks, and (ii) capture the trade-off between the benefit of higher level of care units and needlessly utilizing these units. Second, can this admission control system be designed to achieve a good performance guarantee? These questions motivate us to investigate *online learning* methods that incorporate *control*.

3.1.1 Main Results and Contributions

Our main methodological contributions are the (i) introduction of a personalized admission control system model, and (ii) development and analysis of a new class of online learning algorithms for it, which we call the Personalized Admission Control (PAC) algorithm. In the following, we summarize our main results and contributions.

(a) Our posterior sampling-based algorithm can adaptively learn readmission risks with respect to different types of patients and care unit assignments. We partition the finite time horizon into multiple intervals with equal length. At the beginning of each interval, the algorithm updates the care unit assignment probabilities based on the current belief about the expected rewards (non-readmission probabilities) and the prior assignments. During each interval, the algorithm assigns sequentially arriving patients to different care units based on the assignment probabilities computed for that interval. At the end of each interval, the algorithm collects a batch of realized feedback outcomes and updates the current belief about the expected rewards. These features of the algorithm yield a new learning setting that we call *batch learning with delay*. In this setting, the learning must be done through M batches and the feedback of a patient in a batch is not immediately realized after assigning the patient to a care unit. The earliest time a patient's feedback is realized is at the end of the interval in which they arrived; but it is usually realized at the end of a future interval due to feedback delay. In our theoretical analysis, we derive an upper bound on the *batch learning loss* incurred due to learning the expected rewards (see §3.4.4 for a detailed discussion of our results).

(b) We solve a multi-period admission control problem with online learning in which patients have stochastic lengths of stay, the resources (care unit beds) have finite capacity, and they are *reusable*. The reusable nature of hospital beds adds a nontrivial layer of complexity to our problem. We need to account for the lengths of stay of the assigned patients to different care unit beds and capture the effect of care unit placement decisions on capacity. To do so, we design a *policy guide* model (i.e., a linear program) which approximates the effect of lengths of stay on capacity and ensures a trade-off between the benefit of better health outcomes by assigning patients to SDUs or ICUs and the costly use of these high-demand beds. Our *online algorithm* judiciously makes online care unit placement decisions by leveraging the policy guide model, which is updated after each time interval. Our system model induces a loss network system in which there is a possibility that a patient gets blocked and cannot receive treatment in the assigned care unit. We analyze the *blocking loss* (i.e., the loss due to the possibility of assigning a patient to a fully utilized care unit) and capture the effect of blocking on the theoretical performance of our algorithm.

(c) Our performance measure is *Bayesian regret*, which is the expected loss of an *online policy* compared to an *optimal clairvoyant policy*. On a high level, our algorithm includes two interacting layers for contextual learning and online allocation of reusable resources. Accordingly, it admits a Bayesian regret bound which comes from two major types of loss: (i) the loss associated with contextual learning (batch learning loss), and (ii) the loss associated with the allocation of reusable resources (resource allocation loss). Analyzing the Bayesian regret of our algorithm necessitates: (i) establishing a new high-probability confidence bound on expected rewards (Proposition III.1), (ii) deriving a high-probability upper bound on the batch learning loss (Proposition III.3), (iii) analyzing the *blocking loss* and providing an upper bound on it (Proposition III.4), and (iv) introducing a set of bridging techniques (Theorem III.1) to decompose the Bayesian regret and derive an upper bound on it.

(d) We collaborated with a partner hospital to assess and enhance the real-world applicability of our algorithm using hospital system data. This work provides insight into the potential ability of learning algorithms to reduce readmission rates. From the operational and clinical perspective, our optimization-learning methodology provides a proof of concept for the use of this type of methodology for care unit placements in hospitals. In our method, the information revealed for prior care unit placement decisions is used to reduce the exposure of patients to less effective decisions and explore promising care unit placements. Furthermore, our general method can also deliver cutting-edge methodology to several other applications, including but not limited to computing platforms such as Amazon Web Service (AWS), hospitality services such as Airbnb, and hotel-booking platforms.

3.1.2 Literature Review

Our work is related to the following two streams of literature.

Allocation of Reusable Resources. The problem of allocating reusable resources *without* online learning has been studied in several application domains, including admission control, advance reservation, pricing, and assortment optimization. Several studies have been conducted on admission control and scheduling policies in hospitals. [98] developed a model for maximizing the number of lives saved by investigating variations of first-come-first-served policy to admit patients to the ICU. [60] proposed analytical models to coordinate elective admissions with other hospital subsystems and reduce hospital congestion. [69] conducted an econometric analysis to estimate the cost of denying ICU admission and provided a simulation framework to evaluate the performance of several admission strategies. [96] proposed an average cost dynamic program to optimize patient admissions in a neurology ward with multiple types of patients. [44] proposed a data-driven approach to study the effect of off-service placement on patient outcomes, bed assignment decisions, and the network structure of care units. [40] developed an approximate dynamic programming approach to optimize the allocation of patients to primary and non-primary units. Our work fits into this literature because we develop a *data-driven policy* to make care unit placement decisions. Our policy captures stochastic lengths of stay and availability of limited care unit beds, which are reusable resources. A key feature of our policy is improving the care unit placement decisions on the fly by *adaptively learning* patient outcomes.

We model the hospital admission control as a stochastic control problem. Many stochastic control problems have a corresponding fluid model that yields a deterministic control problem that can often be solved directly as a linear program. The use of fluid models is motivated by the extensive theory of optimal control of deterministic systems. Fluid approximations are frequently used in analyzing stochastic queueing networks with time-varying arrival rates ([80], [16] and [30]), restless bandit ([109] and [23]), admission control ([75]), advance reservation ([34]), and revenue management ([74] and [88]). To design our care unit placement policy, we follow the idea of fluid approximations by introducing a deterministic linear program as a policy guide.

Next, we discuss some of the above-mentioned studies that are more relevant to ours. In an infinite horizon setting with a single reusable resource, there are some studies with near-optimal heuristics with constant factor performance guarantees. [75] studied an admission control problem in which a single reusable resource is used to serve multiple classes of customers. Each customer requests a particular set of resources upon arrival and the requests must be accepted or rejected in real-time. [34] studied an advance reservation problem with a non-homogeneous demand rate and a single type of resource. In their setting, each arriving customer submits a service request upon arrival and specifies the start time and end time, then the seller must decide to accept the request or reject it in real-time. There are some other recent studies in a *finite horizon* setting with *multiple* types of reusable resources. [74] developed an asymptotically near-optimal pricing control algorithm under a deterministic service time assumption. They showed that the algorithm can be extended to a more general setting with heterogeneous service time and advance reservation. [88] studied an assortment optimization problem to offer a set of products to each arriving user, where the choice of the users depends on their preferences over the set of products. They developed a policy for assortment optimization of reusable resources under non-stationary Poisson arrivals and exponentially distributed service times, and proved that the policy is guaranteed to obtain $1/e$ fraction of the optimal total expected revenue. Note that all of the above-mentioned studies assumed that the reward/revenue generated by each resource allocation is known to the decision maker a priori. The closest work to ours is the study of [88] in the finite horizon setting; however, they assumed that the revenue rate of allocating a product to a customer is known *a priori* and their policy can be obtained by a *one-shot* optimization method.

Multi-armed Bandits. Multi-armed bandit (MAB) is an online learning framework for making sequential decisions when the effect of each action on the outcome is uncertain. At each step, the agent selects an action from the possible actions and the goal is to maximize the expected cumulative reward obtained from the selected actions. We refer to [99] and [73] for a comprehensive review.

Recently, there has been a growing interest in the development of sequential decision making algorithms in healthcare using MABs, including response-adaptive clinical trial ([7]), healthcare-adherence interventions ([84]), and treatment policies for chronic diseases ([86]). The contextual MAB (CMAB) is a particularly useful class of MABs where the reward of each action depends on the context that can be observed at each round. In this setting, the agent adaptively collects information and learns the relation between observed information and rewards to select the best action. There is a vast and growing body of literature on CMAB. [8] introduced the first algorithm for the linear CMAB, called LinRel. Afterward, this algorithm was improved by several other studies (e.g., [41], [94], [37], and [1]). In the generalized linear model (GLM) setting, [50] proposed an upper confidence bound (UCB) based algorithm and derived a regret bound for it, which was improved by [77]. Recently, [14] provided a CMAB algorithm, called a LASSO bandit, in a setting where the covariates are high-dimensional. [15] proved that a greedy algorithm can be rate optimal for a two-armed bandit as long as a condition on covariates (i.e., covariate diversity) is met. They also proposed a Greedy-First algorithm that performs exploration when the observed data indicate its necessity. [13] proposed a contextual bandit with cross-learning in which the learner also learns the reward that would have been achieved by choosing the same action under different contexts. [46] proposed a CMAB algorithm that leverages a bootstrapping approach to guide the exploration-exploitation trade-off. [36] introduced a non-stationary bandit by leveraging a combination of stochastic and adversarial bandits. Th previously mentioned studies do not consider the need for decision making under limited resources. However, we must deal with limited resources in our problem setting.

The MAB problem with a Knapsack (BwK) is an important class of MABs where each arm consumes a certain amount of the available resources. The studies of [11] and [4] were among the first to propose a MAB with resource constraints. Afterward, [12], [5], and [3] studied extensions of BwK with independent and identically distributed context vectors. Note that all of the aforementioned studies assume the existence of a *global* knapsack constraint where the term global indicates that the total budget/capacity is fixed and time-invariant. In contrast, we develop an online algorithm under the presence of resources that can be repeatedly used over the planning horizon. This involves developing several new modeling and technical ideas on batch learning under the presence of reusable resources and delayed feedback.

Lastly, there are a few studies in the literature on online service platforms where there is a need for online learning and queueing (see, e.g., [21], [62], [63], and [97]). [62] and [63] studied a setting where there are a fixed number of servers with known service rates. Users arrive at the system with unknown arrival rates, each of which brings a certain number of tasks. All tasks should be assigned to servers and they yield user-server-dependent random rewards. The system aims to maximize the expected rewards. By focusing on the steady-state behavior, they designed algorithms that have distinct phases for exploration and exploitation. Although these studies involve learning when there are limited servers, their algorithms and analyses cannot be extended to our setting because we have non-stationary arrival rates and do not allow for queues in the system. [71] studied the queueing bandit problem, in which each arm is a server that can serve a waiting job. An arriving job should be assigned to a server and there is an unknown success probability. If a job is successfully completed, it departs the system; if it fails, it remains in the queue until it is successfully served. The stochastic reward is a binary value depending on whether the job was successfully served or not. The aim is to minimize the queue-regret. They developed algorithms for a setting with a single queue and multiple servers. In a follow-up study, [72] extended the previous analysis to multiple queues and matching constraints. The algorithms and analyses developed for this problem cannot be extended to ours because we pursue a different aim than minimizing the length of queues.

3.1.3 Organization and General Notation

The remainder of this chapter is organized as follows. We formulate our problem in §3.2 and introduce our online algorithm in §3.3. We carry out a non-asymptotic regret analysis in §3.4. In §3.5, we provide a case study as proof of concept using hospital system data. Finally, we conclude this chapter in §3.6.

All vectors are column vectors. For any column vector $x \in \mathbb{R}^n$, x' denotes its transpose and $[x]_\ell$ indicates its ℓ^{th} element. The determinant and trace of a square matrix M is denoted by $\det(M)$ and $\operatorname{tr}(M)$. Also, I denotes the identity matrix. The Euclidean norm and weighted norm of x are denoted by $\|x\| = \sqrt{x'x}$ and $\|x\|_M = \sqrt{x'Mx}$, respectively. For two symmetric matrices A and B, $A \succeq B$ ($A \succ B$) means that $A - B$ is positive semidefinite (positive definite). For a symmetric positive definite matrix V, we define $\rho_{\min}(V)$ as the smallest eigenvalue of V. We use $\mathbb{1}(\cdot)$ as the indicator function. We follow the convention that $\sum_{s=i}^{j} a_s = 0$ if $i > j$.

3.2 Personalized Admission Control System Model

We formally define our personalized admission control system model. We consider a finite and discrete time horizon. We partition the time horizon into M disjoint and fixed-length intervals and denote $\mathcal{M} = \{1, \ldots, M\}$ as the set of disjoint intervals. Let $\mathcal{J} = \{1, \ldots, J\}$ be a set of care units and $\mathcal{K} = \{1, \ldots, K\}$ be a set of patient types. Let $N_k(m)$ be the number of arrivals of type $k \in \mathcal{K}$ patients during interval m. It follows a Poisson distribution with known mean $\lambda_k(m)$. We define $\bar{N}(m) = \sum_{k=1}^{K} N_k(m)$ and $\bar{\lambda}(m) = \sum_{k=1}^{K} \lambda_k(m)$.

Upon arrival, each patient is associated with a context vector (e.g., age, gender, ethnicity, medical history) and needs to be assigned to a care unit. Let $\varphi_k \in \Xi$ be the *context vector* of any patient of type $k \in \mathcal{K}$. As is standard in the literature, we assume the existence of a known feature map $\Lambda : \Xi \times \mathcal{J} \to \mathbb{R}^d$ and we define $\phi_{kj} = \Lambda(\varphi_k, j)$ as the d-dimensional *feature vector*. To identify a patient's feature vector based on their order of arrival during an interval, it is notationally simpler to denote this term as $\phi_j^{(i)}(m)$ where i refers to the order of arrival, and i and m correspond to a single patient with a specific type k. Hence, $\phi_j^{(i)}(m)$ uniquely identifies the feature vector of the i^{th} patient who arrived at interval m and is assigned to care unit j.

Lengths of Stay. A hospital always has a limited number of beds (*reusable resources*) in the intensive and intermediate care units. A patient should be assigned to a care unit upon arrival. After a random amount of time, the patient gets discharged and the occupied bed becomes available again. In our setting, we assume that the length of stay (LOS) is exponentially distributed with known rate μ_{kj} for a type k patient assigned to care unit j. This is a common assumption in the literature for tractability (see, e.g., [98], [69], and [88]).

Patient Reward. Each action/care unit assignment yields an uncertain binary reward, where 1 corresponds to success (i.e., patient is *not-readmitted*) and 0 corresponds to failure (i.e., patient is *readmitted*). The choice of care unit $j \in \mathcal{J}$ for the i^{th} patient who arrived at interval m yields the following stochastic reward:

$$\mathcal{R}_j^{(i)}(m) = \sigma(\langle\, \phi_j^{(i)}(m), w\rangle) + \xi_j^{(i)}(m),$$

where $w \in \mathbb{R}^d$ is the unknown (true) model parameter and $\sigma(y) = (1 + e^{-y})^{-1}$ for $y \in \mathbb{R}$

is the logistic function. The noise values, $\xi_j^{(i)}(m)$, are zero-mean independent and bounded random variables. To simplify the notation, we let $r_j^{(i)}(m) = \sigma(\langle \phi_j^{(i)}(m), w\rangle)$ be the expected reward associated with the i^{th} patient who arrived at interval m and was assigned to care unit j under the parameter w.

Our admission control system model receives the readmission outcome as the feedback, which cannot be realized immediately after assigning a patient to a care unit. In particular, a non-readmission event cannot be realized until a certain number of days after the discharge date (e.g., 30 days in our case study) and a readmission event cannot be realized until the day of readmission.

Care Unit Placement Policy. The problem can be modeled as a discrete-time and finite-horizon Markov decision process (MDP). By the memoryless property of arrival process and lengths of stay, the *current* occupancy of beds in different care units is enough to know the state of the system. The state of our system, $u \in \mathcal{U}$, is observed at the beginning of an interval and can be defined by the number of patients of type $k \in \mathcal{K}$ in care unit $j \in \mathcal{J}$ (i.e., each state can be specified by a (JK)-dimensional vector). We define a *decision rule* at interval m for type k patients as a mapping from the state of the system to a distribution over the set of care units $\mathcal{J}$:

$$\pi_m^k : \mathcal{U} \to \Omega\ (\mathcal{J}),$$

where $\Omega\ (\mathcal{J})$ is the space of probability distributions over the set $\mathcal{J}$. Note that we use $\pi_m^k(j|u)$ to denote the probability of offering care unit j to a type k patient given the current state of the system, $u \in \mathcal{U}$, at interval m.

Accordingly, a *state-dependent policy* $\pi \in \Pi$ can be formally defined as a sequence of decision rules π_m^k for $m \in \mathcal{M}$ and $k \in \mathcal{K}$. Due to the curse of dimensionality and the limited information setting in our problem, we design a state-independent policy for care unit assignments that performs well compared to the optimal state-dependent policy.

The goal of our personalized admission control system model is to maximize the total expected reward over a finite time horizon. For technical reasons, we denote $D_{\max}$ as the maximum number of intervals required for a feedback to be realized, and let $\bar{N}_{\max} = \max_{m \in \mathcal{M}} \bar{N}(m)$. We also define $T = \sum_{m=1}^{M} \bar{N}(m)$ and $\bar{T} = \sum_{m=1}^{M} \bar{\lambda}(m)$. To simplify the presentation of our results, we assume $\bar{N}_{\max} \leq \zeta \, \frac{T}{M}$ where ζ is a positive constant that regulates the maximum deviation of the number of arrivals across different intervals. Without loss of generality, we assume $\|\phi_{kj}\| \leq 1$ and $\|w\| \leq 1$.

3.2.1 Benchmark and Linear Programming Relaxation

The notion of *regret* is commonly used in the literature as a metric to evaluate the theoretical performance of a policy when the decision maker has limited access to information. The regret of a policy is the average difference between the total reward obtained by the policy and the total reward obtained by a benchmark, which is an optimal policy that has access to the *full information*. In our problem, the full information setting corresponds to the knowledge of the expected reward with respect to the assignment of each type of patient to each care unit. This information is accessible when the model parameter w is known. Our goal is to find an assignment policy for the limited information setting to maximize the total expected reward and thereby yield a small regret over a finite time horizon.

For our problem, computing an *optimal state-dependent policy* under the full information setting is intractable due to the curse of dimensionality. As an alternative, we propose a *deterministic linear program* that yields *state-independent* assignment probabilities, and we prove that the optimal objective value of this linear program is an *upper bound* on the optimal expected reward achieved by the optimal state-dependent policy. The deterministic linear program can be created by following a popular approach, called fluid approximation, to approximate the decision problem and reduce it into a simpler optimization problem (see, e.g., [51] and [79]). To relax the original problem, our deterministic linear program enforces the capacity constraints only in expectation, while the capacity constraints have to hold for every sample path in the original problem.

In the following, we first characterize the state-dependent optimal policy. Then, we introduce our deterministic linear program formulation. Finally, we show that the optimal expected reward obtained by the optimal state-dependent policy is upper bounded by the optimal objective value obtained by the deterministic linear program.

We restrict the space of policies to the admissible policies that are (i) non-anticipative (i.e., only use the information revealed up to the current time except the arrival rate information across all intervals), and (ii) feasible (i.e., the total number of patients assigned to each care unit does not exceed the capacity limit). We denote Π as the space of all admissible policies. For an admissible policy $\pi \in \Pi$, we define $\Theta^{\pi}_{kj}(m)$ as the number of type k patients in care unit j at the beginning of interval m. Because of the capacity constraints, we have $\sum_{k=1}^{K} \Theta^{\pi}_{kj}(m) \leq C_j$ for all $j \in \mathcal{J}$ and $m \in \mathcal{M}$, where C_j is the capacity (total number of beds) of care unit j. Let $\beta^{\pi}_{kj}(m)$ denote the number of type k patients who arrived to care unit j during interval m. Also, let $D^{\pi}_{kj}(m)$ denote the number of departures of type k patients from care unit j during interval m. We observe that the following flow balance constraints hold for any $j \in \mathcal{J}$ and $m \in \mathcal{M}$:

$$\sum_{k=1}^{K} \Theta_{kj}^{\pi}(m) + \sum_{k=1}^{K} \beta_{kj}^{\pi}(m) \leq C_j + \sum_{k=1}^{K} D_{kj}^{\pi}(m), \tag{3.1}$$

$$\Theta_{kj}^{\pi}(m+1) = \Theta_{kj}^{\pi}(m) + \beta_{kj}^{\pi}(m) - D_{kj}^{\pi}(m). \tag{3.2}$$

For each interval $m \in \mathcal{M}$, any policy $\pi \in \Pi$ induces a distribution $\mathcal{P}_m^{\pi}(u)$ over the possible occupancy states, which captures the randomness in the arrival process, the lengths of stay, and the possible randomization in the policy. We define $x_{kj}^{\pi}(m)$ as the probability that policy π assigns a type k patient to care unit j during interval m, which can be obtained as follows:

$$x_{kj}^{\pi}(m) \;=\; \sum_{u \in \mathcal{U}} \mathcal{P}_m^{\pi}(u)\; \pi_m^{k}(j|u).$$

Accordingly, the *average* number of type k patients arriving to care unit j during interval m is calculated by $\bar{\beta}_{kj}^{\pi}(m) = \lambda_k(m)\, x_{kj}^{\pi}(m)$. Let V^{π} be the total reward obtained by policy $\pi \in \Pi$ given w. Then, it has the following conditional expectation:

$$\mathbb{E}[V^{\pi}|w] = \sum_{m=1}^{M} \sum_{k=1}^{K} \sum_{j=1}^{J} r_{kj}\, \bar{\beta}_{kj}^{\pi}(m),$$

where $r_{kj} = \sigma(\langle\, \phi_{kj}, w\rangle)$ is the expected reward of a type $k \in \mathcal{K}$ patient assigned to care unit $j \in \mathcal{J}$.

Next, we define V^{π^*} as the optimal (total) reward obtained by the optimal state-dependent policy π^* given w. Then, we have:

$$\mathbb{E}[V^{\pi^*}|w] = \sup_{\pi \in \Pi} \mathbb{E}[V^{\pi}|w].$$

To formulate our deterministic linear program, we need to compute the probability that a patient remains in the care unit assigned until the end of a subsequent interval. Assume that a type k patient is assigned to care unit j at the beginning of interval m, then the probability that the patient remains in the respective care unit until the end of the interval is $e^{-\mu_{kj}}$. It follows that the probability of a type k patient, whose arrival happens during interval s and who is assigned to care unit j, to be still in the bed at the end of interval m can be lower bounded by:

$$\psi_{kj}(s,m) = \begin{cases} e^{-(m-s+1)\,\mu_{kj}} & \text{if } s \leq m \\ 0 & \text{if } s > m. \end{cases}$$

Now, we are ready to formulate our deterministic linear program. Given the model parameter w, we define it as follows:

Deterministic Linear Program ($\boldsymbol{LP^{UB}}$)

$$\max_{x} \quad \sum_{m=1}^{M}\sum_{k=1}^{K}\sum_{j=1}^{J} r_{kj}\ \lambda_k(m)\ x_{kj}(m) \tag{3.3}$$

$$\text{s.t.} \quad \sum_{s=1}^{m}\sum_{k=1}^{K} \lambda_k(s)\ x_{kj}(s)\ \psi_{kj}(s,m) \ \leq \ C_j, \quad \forall\, j \in \mathcal{J},\ \forall\, m \in \mathcal{M} \tag{3.4}$$

$$\sum_{j=1}^{J} x_{kj}(m) \ \leq \ 1, \quad \forall\, k \in \mathcal{K}, \quad \forall\, m \in \mathcal{M} \tag{3.5}$$

$$x_{kj}(m) \ \geq \ 0, \quad \forall\, k \in \mathcal{K},\ \forall\, j \in \mathcal{J},\ \forall\, m \in \mathcal{M}. \tag{3.6}$$

The objective function (3.3) maximizes the expected reward over the finite time horizon. Constraint (3.4) ensures that the average number of patients assigned to a care unit up to any interval does not exceed the total capacity of the care unit. Constraint (3.5) specifies that the sum of probabilities of choosing a care unit for each patient type cannot exceed one.

Let $x^*_{kj}(m)$ be the optimal solution of the above linear program, which denotes the probability of assigning a type k patient to care unit j during interval m. Next, Lemma III.1 proves that, given w, $OPT^{UB} = \sum_{m=1}^{M}\sum_{k=1}^{K}\sum_{j=1}^{J} r_{kj}\ \lambda_k(m)\ x^*_{kj}(m)$ provides an *upper bound* on the optimal expected reward achieved by the optimal state-dependent policy $\mathbb{E}[V^{\pi^*}|w]$. We highlight that the assignment probabilities obtained by the above linear program are state-independent and the obtained $x^*_{kj}(m)$ may yield infeasible solutions since LP^{UB} enforces the capacity constraints only in expectation. Thus, the above linear program can be viewed as a relaxed benchmark.

Lemma III.1 (Upper Bound on Optimal Expected Reward). *Let OPT^{UB} be the optimal objective value of LP^{UB} given the model parameter w. Then, the optimal expected reward achieved by the optimal state-dependent policy $\mathbb{E}[V^{\pi^*}|w]$ is upper bounded by OPT^{UB}.*

The proof of Lemma III.1 can be found in Appendix A.

3.3 Personalized Admission Control Algorithm

In this section, we propose an online algorithm that leverages online learning and optimization techniques and provides a personalized admission control system. In §3.3.1, we describe the high-level intuition of our algorithm. We then provide the detailed steps of the personalized admission control (PAC) algorithm in §3.3.2.

3.3.1 Main Ideas of the Personalized Admission Control Algorithm

In our problem, each type of patient can be characterized by a unique set of characteristics (e.g., medical records, diagnostics tests, and demographic information), and there is uncertainty about the readmission impact of a care unit placement decision. This decision should be made based on both patient characteristics and congestion in different care units with limited beds. Also, it should capture the trade-off between the benefit of having a lower risk of readmission in a higher level bed versus the opportunity cost of not having this high-demand bed available when a more complex patient arrives in the future. The PAC algorithm is designed to provide care unit placement decisions by synergizing contextual learning and online optimization techniques, and it aims to maximize the expected reward (minimize the expected readmission risk) over a finite time horizon.

There are two main challenges in deciding about care unit placements: (i) *unknown* patient rewards, and (ii) accounting for *limited reusable* resources. In what follows, we explain the intuition behind designing the PAC algorithm to address these challenges.

Suppose that we know the expected reward of each type of patient with respect to each care unit assignment. Thus, the remaining challenge is to design a mechanism to make care unit placement decisions. A possible care unit placement mechanism would be a *greedy mechanism*, in which each patient should be assigned to a care unit that yields the highest expected reward for that patient. It is well known that this may not result in obtaining a high total expected reward over a time horizon, because it *ignores* the effect of current decisions on the subsequent ones and does not capture the trade-off between the benefit of higher level of care units and needlessly utilizing the limited beds in these units.

To overcome this greediness and maximize the expected reward over the entire time horizon, we use an optimization method. As we discussed in §3.2.1, an optimal state-dependent policy achieves the optimal expected reward, but the curse of dimensionality limits its usefulness. A common and powerful strategy is to relax the problem by enforcing the capacity constraints only in expectation. Recall that LP^{UB} proposed in §3.2.1 provides an upper bound on the optimal expected reward obtained by the optimal state-dependent policy. This motivates us to adopt such a linear program to make care unit placement decisions. Note that LP^{UB} relaxes the problem and generates a state-independent assignment probability for each type of patient with respect to each time interval. The cost of this relaxation is the possibility of blocking, which happens when a patient is assigned to a care unit with no available bed. To take advantage of such an intuitive and easy-to-implement approach for care unit placement, we propose a variation of LP^{UB} that provides a good relaxation for which the loss that the system might incur due to blocking is upper bounded and under control. We define a linear program called deterministic linear program with capacity buffer (LP^{C-UB}), which is identical to LP^{UB}, except that the capacity of care units are scaled down by a pre-specified multiplier. This ensures that the loss due to the possibility of blocking is limited and under control. We formally define LP^{C-UB} as follows:

Deterministic Linear Program with Capacity Buffer ($\boldsymbol{LP^{C-UB}}$)

$$\max_{x} \quad \sum_{m=1}^{M} \sum_{k=1}^{K} \sum_{j=1}^{J} r_{kj} \; \lambda_k(m) \; x_{kj}(m) \tag{3.7}$$

$$\text{s.t.} \quad \sum_{s=1}^{m} \sum_{k=1}^{K} \lambda_k(s) \; x_{kj}(s) \; \psi_{kj}(s,m) \;\; \leq \;\; e^{-2 \max\limits_{k,j} (\mu_{kj})} \, C_j, \quad \forall\, j \in \mathcal{J}, \;\; \forall\, m \in \mathcal{M} \tag{3.8}$$

$$\sum_{j=1}^{J} x_{kj}(m) \;\; \leq \;\; 1, \quad \forall\, k \in \mathcal{K}, \quad \forall\, m \in \mathcal{M} \tag{3.9}$$

$$x_{kj}(m) \;\; \geq \;\; 0, \quad \forall\, k \in \mathcal{K}, \;\; \forall\, j \in \mathcal{J}, \;\; \forall\, m \in \mathcal{M}, \tag{3.10}$$

where $x_{kj}(m)$ can be interpreted as the probability of assigning a type k patient to care unit j during interval m.

In Lemma III.2, we prove that adding this capacity buffer reduces the resulting objective function by at most a constant ratio compared to OPT^{UB} (i.e., $OPT^{C-UB} \geq e^{-2\max\limits_{k,j}(\mu_{kj})} \, OPT^{UB}$). Thus, it shows that LP^{C-UB} provides a time interval- and type-dependent policy for care unit placements that is state-independent while obtaining a high fraction of the expected reward obtained by LP^{UB}.

Lemma III.2 (**Lower Bound on Optimal Objective Value of LP^{C-UB}**)**.** *Let OPT^{C-UB} be the optimal objective value of LP^{C-UB} given the model parameter w. Then, we have $OPT^{C-UB} \geq e^{-2\max\limits_{k,j}(\mu_{kj})} \, OPT^{UB}$.*

The proof of Lemma III.2 can be found in Appendix A.

Next, we discuss the challenge that the expected rewards of patients are not known in our problem. Our algorithm should learn the expected reward (i.e., the probability of non-readmission) of each patient type with respect to assignment to each care unit. We take advantage of the structure of the reward model that allows for our algorithm to transfer learning from one patient type to another. In particular, the expected reward of a patient depends on a feature vector and an unknown vector parameter that needs to be learned based on the available information. This structure allows the algorithm to work in a high-dimensional setting with a large number of patient types and care units. It is worth noting that in a low-dimensional setting, it is also possible to separately learn the reward of each type of patient and action/care unit assignment pair, but this method is not efficient in a high-dimensional setting because it does not allow for information sharing.

The learning process suffers from bandit feedback, meaning that we only obtain feedback from the selected decision and we do not observe counterfactual rewards for alternative decisions. This hurdle may result in locking into a misperception caused by a lack of data. For instance, based on our uncertain estimates in the early stage, we may incorrectly conclude that SDU is not a good fit for a type of patient with a certain history of disease and discard it for this type of patient. Subsequently, we may not be able to identify this incorrect belief since we will not observe counterfactual rewards. To avoid this, there should be a thoughtful trade-off between exploration and exploitation. Inspired by the idea of *posterior sampling*, our algorithm assumes a posterior distribution over the unknown model parameter, and then takes random samples from the posterior distribution. The intuition behind this sampling is to balance the *exploration-exploitation trade-off*. If the algorithm only used the mean of the posterior distribution in each time interval as an estimate for the unknown vector parameters, it would exploit the current belief about the unknown parameter, and there would be insufficiently exploring alternative choices. Therefore, we take random samples from the posterior distribution to carry out explorations.

In our setting, we collect a new batch of feedback outcomes after each interval that can be used to improve the estimated expected rewards. In particular, the learning should be done through M batches and the feedback of a patient in a batch is not immediately realized after assigning the patient to a care unit. In our problem, a non-readmission event cannot be realized until a certain number of days after the discharge date and a readmission event cannot be realized until the day of readmission. This is contrary to the typical online learning problems in which the feedback is realized immediately and the estimator is updated after receiving each feedback. This brings us to a new learning setting that we call *batch learning with delay.* Since we need to learn from the realized feedback outcomes gradually, we cannot follow a one-shot optimization method by solving LP^{C-UB}. Instead, our algorithm solves LP^{C-UB} iteratively as a *policy guide* model. That is, at the beginning of each interval, the algorithm obtains the assignment probabilities for the current interval by resolving LP^{C-UB} using the updated beliefs and the assignment probabilities obtained in the prior iterations.

3.3.2 Description of the Personalized Admission Control Algorithm

Let $j^{(i),Alg^*}(m)$ be the selected care unit by PAC for the i^{th} patient who arrived at interval m, which we will often write as j^* when indices (i, m) are obvious. We also let $D^{(i)}(m)$ be the feedback delay of the i^{th} patient who arrived at interval m and let (m, i, j^*) be a tuple referring to the i^{th} patient who arrived at interval m and was assigned to care unit j^*. We define $\mathcal{F}(m) = \big\{(s, i, j^*)\,|s + D^{(i)}(s) \leq m - 1;\, i \in \{1, \ldots, \bar{N}(s)\}\big\}$ as a set that contains tuples (s, i, j^*) of patients with *realized* readmission feedback outcomes up to the end of interval $m-1$. We also define a mapping function $\chi : \{0,1\} \to \{-1,1\}$, where $\chi(0) = -1$ and $\chi(1) = 1$. The detailed steps of the online PAC algorithm are provided in Algorithm 1.

Description. In the first step, the parameters p_ℓ^1 and $(q_\ell^1)^{-1}$ are initialized based on some prior beliefs. The algorithm proceeds in M intervals. At the beginning of each interval, we need to obtain the assignment probabilities by solving the LP at Step (5), in which we use samples drawn from the posterior distributions obtained using the collected information up to the beginning of the interval. The objective function of this LP is to maximize the expected reward over the horizon of subsequent intervals beginning with the current one. The first constraint ensures that the average number of patients assigned to each care unit does not exceed the available capacity with buffer. The second constraint specifies that the sum of probabilities for choosing a care unit for each patient type cannot exceed one. After solving the LP, we obtain $x_{kj}(m)$ which is the probability of assigning a patient of type k to care unit j during interval m. Then, we record the assignment probabilities corresponding to the current interval that will be used in the subsequent intervals.

Next, we assign arriving patients of type k to care unit j based on $x_{kj}(m)$. After deciding about the care unit placement of all patients who arrived during the current interval, we use the realized batch of feedback outcomes to update the posterior distributions. The updating process is done using an online Bayesian regression based on a Laplace approximation (see [108] for more information). In particular, at the end of interval m, we update the parameters of posterior distributions *on the fly* at Steps (9)-(12) using set $\mathcal{B}(m+1)$, which contains tuples (s, i, j^*) of patients for which their feedback outcomes are *realized* during interval m.

3.4 Theoretical Performance Analysis and Discussions

We derive a non-asymptotic (i.e., finite-time) performance guarantee for the PAC algorithm. We start by defining the performance measure in §3.4.1. Next, we provide a roadmap for our regret analysis in §3.4.2. Then, we state our main theoretical results in §3.4.3. Finally, we discuss the results in §3.4.4.

3.4.1 Performance Measure

We use the *Bayesian regret* as a metric to evaluate the theoretical performance of our algorithm. The regret of an algorithm is its cumulative loss relative to a benchmark. The Bayesian regret is the expected regret, where the expectation is taken with respect to the prior distribution of the unknown model parameter (see Definition III.1). This quantity is called Bayesian regret since it represents the *Bayes risk* (see, e.g., [94] and [95]). The Bayesian regret has two main advantages: (i) it allows for an arbitrary prior distribution

Algorithm 1 PAC Algorithm

```
Initialize $p_\ell^1$ and $(q_\ell^1)^{-1}$ as the mean and variance of the Gaussian prior distribution over $[w]_\ell$, respectively.
for $m = \{1, \dots, M\}$ do
    Sample $[\tilde{w}(m)]_\ell$ from the posterior distribution $\mathcal{N}\big(p_\ell^m, (q_\ell^m)^{-1}\big), \ \forall \ell \in \{1, \dots, d\}$.
    Set $\tilde{r}_{kj}(\underline{m}) = \sigma\big(\langle \phi_{kj}, \tilde{w}(m)\rangle\big), \ \forall k \in \mathcal{K}, \ \forall j \in \mathcal{J}, \ \forall \underline{m} \in \{m, \dots, M\}$.
    Solve the LP with decision variables $x_{kj}(\underline{m}), \ \forall k \in \mathcal{K}, \ \forall j \in \mathcal{J}, \ \forall \underline{m} \in \{m, \dots, M\}$:

        $\max_x \ \sum_{s=m}^{M} \sum_{k=1}^{K} \sum_{j=1}^{J} \tilde{r}_{kj}(s) \ \lambda_k(s) \ x_{kj}(s)$
        $\text{s.t.} \ \sum_{s=m}^{\underline{m}} \sum_{k=1}^{K} \lambda_k(s) \ x_{kj}(s) \ \psi_{kj}(s, \underline{m}) \ \leq \ e^{-2 \max_{k,j}(\mu_{kj})} \ C_j - U_j(m, \underline{m}),$
        $\forall j \in \mathcal{J}, \ \forall \underline{m} \in \{m, \dots, M\}$
        $\sum_{j=1}^{J} x_{kj}(\underline{m}) \ \leq \ 1, \quad \forall k \in \mathcal{K}, \quad \forall \underline{m} \in \{m, \dots, M\}$
        $x_{kj}(\underline{m}) \ \geq \ 0, \quad \forall k \in \mathcal{K}, \ \forall j \in \mathcal{J}, \ \forall \underline{m} \in \{m, \dots, M\},$

    where $U_j(m, \underline{m}) = \sum_{s=1}^{m-1} \sum_{k=1}^{K} \lambda_k(s) \ x_{kj}(s) \ \psi_{kj}(s, \underline{m})$ is a constant in the optimization since all $x_{kj}(s)$ for $1 \leq s \leq m-1$ have already been determined in previous iterations.
    Record and fix $x_{kj}(m), \ \forall k \in \mathcal{K}, \ \forall j \in \mathcal{J}$.
    for each arriving patient during interval $m$ do
        Assign patient of type $k$ to care unit $j$ with probability of $x_{kj}(m)$.
    Obtain set $\mathcal{B}(m+1) = \mathcal{F}(m+1) - \mathcal{F}(m)$.
    Solve the following optimization problem:

        $\eta^{\max} = \arg\max_{\eta} \ \frac{1}{2} \sum_{\ell=1}^{d} q_\ell^m([\eta]_\ell - p_\ell^m)^2 + \sum_{(s,i,j^*) \in \mathcal{B}(m+1)} \log\big(1 + e^{-\chi(\mathcal{R}_{j^*}^{(i)}(s)) \, \langle \eta, \, \phi_{j^*}^{(i)}(s)\rangle}\big).$

    Set $p^{m+1} = \eta^{\max}$.
    Set $q_\ell^{m+1} = q_\ell^m + \sum_{(s,i,j^*) \in \mathcal{B}(m+1)} \frac{e^{-\langle \eta^{\max}, \, \phi_{j^*}^{(i)}(s)\rangle}}{\big(1 + e^{-\langle \eta^{\max}, \, \phi_{j^*}^{(i)}(s)\rangle}\big)^2} \big([\phi_{j^*}^{(i)}(s)]_\ell\big)^2, \ \forall \ell \in \{1, \dots, d\}$.
```

over a particular class of mean reward functions, and (ii) it makes a connection between the posterior sampling and upper confidence bound methods. This provides the opportunity to leverage some of the appealing theoretical properties of the upper confidence bound method in our theoretical analysis.

Definition III.1 (Bayesian Regret). Given the unknown model parameter w, the regret over the finite time horizon with M disjoint and fixed-length intervals is defined by

$$\textsc{Reg}(\bar{T}, w) = \mathbb{E}\big[BM - ALG|w\big],$$

where $\bar{T} = \sum_{m=1}^{M} \bar{\lambda}(m)$. Also, BM and ALG are the total rewards of the benchmark and the algorithm over the finite time horizon, respectively. The conditional expectation is taken over the random arrivals, random realizations given w, and possible randomization in the algorithm (i.e., random samples in posterior sampling and randomization in action selection).

Accordingly, the Bayesian regret can be calculated by taking expectation over the prior distribution of w as follows:

$$\textsc{BayesReg}(\bar{T}) = \mathbb{E}\big[\textsc{Reg}(\bar{T}, w)\big].$$

3.4.2 Roadmap for the Main Theoretical Results

We provide a road-map for proving our main theoretical result (Theorem III.1). We start by defining an *auxiliary problem* that facilitates analyzing the theoretical performance of PAC. The auxiliary problem is similar to our original problem defined in §3.2 with one difference that the unknown model parameter is *known* a priori. Similarly, we define an *auxiliary algorithm* to solve the auxiliary problem, which is similar to PAC except that it does not need to learn the model parameter over time. In particular, the auxiliary algorithm only includes solving LP^{C-UB} and following its solution to assign patients to care units.

For notational convenience, we define $x_j^{(i),Aux^*}(m)$, and $x_j^{(i),Alg^*}(m)$ as the assignment probabilities obtained by the auxiliary algorithm and PAC corresponding to the i^{th} patient who arrived at interval m with respect to care unit j, respectively. We also let $j^{(i),Alg^*}(m)$ be the selected care unit by the auxiliary algorithm for the i^{th} patient who arrived at interval

m. Next, we define the following notation:

$$V^{Aux} = \sum_{m=1}^{M} \sum_{i=1}^{\bar{N}(m)} \sum_{j=1}^{J} r_j^{(i)}(m)\ \mathbb{1}\{j^{(i),Aux^*}(m) = j\},$$
$$V^{Alg} = \sum_{m=1}^{M} \sum_{i=1}^{\bar{N}(m)} \sum_{j=1}^{J} r_j^{(i)}(m)\ \mathbb{1}\{j^{(i),Alg^*}(m) = j\},$$

where (i) V^{Aux} is the total reward obtained by the *auxiliary algorithm*, which knows the model parameter w in advance and (ii) V^{Alg} is the total reward obtained by our algorithm, which does *not* know the model parameter w a priori. Note that there is a possibility for patients to get blocked when we assign patients following the assignment probabilities obtained by our algorithm or the auxiliary algorithm, and the loss due to the blocking is not taken into account in V^{Alg} and V^{Aux}. We let V^{BL} denote the loss due to the blocking in the PAC algorithm. Thus, the actual total reward obtained by PAC can be calculated by $V^{Alg} - V^{BL}$.

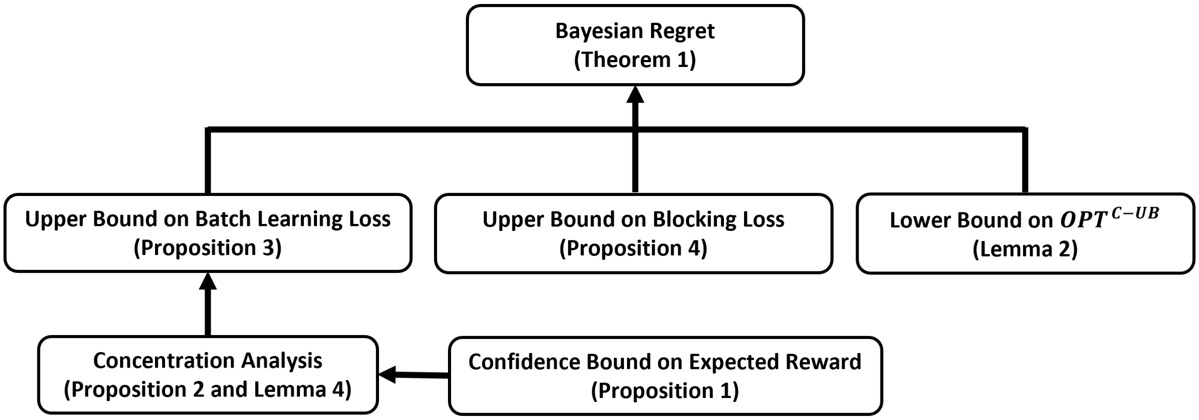

Bayesian Regret
(Theorem 1)
Upper Bound on Batch Learning Loss
(Proposition 3)
Upper Bound on Blocking Loss
(Proposition 4)
Lower Bound on OPT^{C-UB}
(Lemma 2)
Concentration Analysis
(Proposition 2 and Lemma 4)
Confidence Bound on Expected Reward
(Proposition 1)

Figure 3.1: Outline for deriving the Bayesian regret of the PAC algorithm.

Figure 3.1 outlines the main steps for deriving the Bayesian regret of the PAC algorithm. Proposition III.1 establishes a high-probability confidence bound on the expected reward. Lemma III.4 and Proposition III.2 establish upper bounds on two main terms that are essential to calculate the batch learning loss. Proposition III.3 provides an upper bound on the batch learning loss. Lemma III.2 proves a lower bound on the optimal objective value obtained by LP^{C-UB}. Proposition III.4 derives an upper bound on the loss due to blocking. Finally, we bound the Bayesian regret of our algorithm in Theorem III.1 using the direct results of Propositions III.3, III.4, and Lemma III.2.

3.4.3 Main Theoretical Results

Below, we state our main theoretical result.

Theorem III.1 (**Bayesian Regret of the PAC Algorithm**). *With $\delta = \frac{1}{\bar{T}}$, the Bayesian regret of the PAC algorithm over the finite time horizon with M disjoint and fixed-length intervals is upper bounded with probability at least $3(1-\frac{1}{\bar{T}})^3$ as:*

$$\textsc{BayesReg}(\bar{T}) \leq \Big(1 - e^{-2\max\limits_{k,j}(\mu_{kj})-1}\Big)\mathbb{E}\big[OPT^{UB}\big] + \frac{1}{e}\,\widetilde{\mathcal{O}}\left(d\sqrt{\bar{T}} + dD_{\max}\frac{\bar{T}}{M}\left(1 + \frac{\bar{T}}{dM}\right)\right),$$

where $\widetilde{\mathcal{O}}(\cdot)$ hides logarithmic factors. Also, $\bar{T} = \sum_{m=1}^{M}\bar{\lambda}(m)$, $\mathbb{E}[OPT^{UB}]$ is the expected value of OPT^{UB} over the prior distribution of the unknown model parameter, and $D_{\max}$ is the maximum number of intervals required for a feedback to be realized.

Proof. We first decompose the Bayesian regret by the following bridging technique:

$$\begin{aligned}\textsc{BayesReg}(\bar{T}) &= \mathbb{E}\big[V^{\pi^*} - \big(V^{Alg} - V^{BL}\big)\big]\\ &\leq \mathbb{E}\big[OPT^{UB} - V^{Alg}\big] + \mathbb{E}\big[V^{BL}\big],\end{aligned}$$

where V^{π^*} is the total reward obtained by the optimal policy and $V^{Alg} - V^{BL}$ is the total reward of PAC by accounting for the blocked patients. The inequality holds by Lemma III.1.

Next, we have the following by Proposition III.4:

$$\mathbb{E}[V^{BL}|w] \leq \left(1 - \min_{j\in\mathcal{J}}\left\{\sum_{n=0}^{C_j-1}\frac{C_j^n e^{-C_j}}{n!}\right\}\right)\mathbb{E}[V^{Alg}|w].$$

By taking expectation over the prior distribution of w on both sides, we have:

$$\begin{aligned}\mathbb{E}\big[V^{Alg} - V^{BL}\big] &\geq \min_{j\in\mathcal{J}}\left\{\sum_{n=0}^{C_j-1}\frac{C_j^n e^{-C_j}}{n!}\right\}\mathbb{E}\big[V^{Alg}\big]\\ &\geq \frac{1}{e}\,\mathbb{E}\big[V^{Alg}\big],\end{aligned} \tag{3.11}$$

where the last inequality holds since $\sum_{n=0}^{C_j-1}\frac{C_j^n e^{-C_j}}{n!} \geq \frac{1}{e}$ for any $j \in \mathcal{J}$.

Based on the definitions of V^{Aux} and V^{Alg} provided in §3.4.2, we have:

$$\begin{aligned}\mathbb{E}\big[V^{Aux} - V^{Alg}\big] &= \mathbb{E}\left[\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\sum_{j=1}^{J} r_j^{(i)}(m)\, x_j^{(i),Aux^*}(m) \;-\; \sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\sum_{j=1}^{J} r_j^{(i)}(m)\, x_j^{(i),Alg^*}(m)\right]\\ &= \sum_{m=1}^{M}\sum_{k=1}^{K}\sum_{j=1}^{J}\mathbb{E}\left[r_{kj}\,\lambda_k(m)\, x_{kj}^{Aux^*}(m) - r_{kj}\,\lambda_k(m)\, x_{kj}^{Alg^*}(m)\right].\end{aligned}$$

The above term can be bounded by the result of Proposition III.3 (see Appendix B). For any $\delta > 0$, the following holds with probability at least $1-\delta$.

$$\sum_{m=1}^{M}\sum_{k=1}^{K}\sum_{j=1}^{J}\mathbb{E}\left[r_{kj}\,\lambda_k(m)\, x_{kj}^{Aux^*}(m) - r_{kj}\,\lambda_k(m)\, x_{kj}^{Alg^*}(m)\right] \le L(\bar{T},\delta).$$

According to (3.11) and the above upper bound, the following holds with probability at least $1-\delta$:

$$\begin{aligned}\mathbb{E}\big[V^{Alg} - V^{BL}\big] &\ge \frac{1}{e}\Big[\mathbb{E}\big[V^{Aux}\big] - L(\bar{T},\delta)\Big]\\ &\ge e^{-2\max\limits_{k,j}(\mu_{kj})-1}\,\mathbb{E}\big[OPT^{UB}\big] - \frac{1}{e}\,L(\bar{T},\delta),\end{aligned}$$

where the second inequality holds by $\mathbb{E}[V^{Aux}|w] \ge e^{-2\max\limits_{k,j}(\mu_{kj})}\, OPT^{UB}$ (see Lemma III.2 in Appendix A).

Finally, using some simple algebra, the following holds with probability at least $1-\delta$:

$$\mathbb{E}\big[OPT^{UB} - V^{Alg}\big] + \mathbb{E}\big[V^{BL}\big] \le \Big(1 - e^{-2\max\limits_{k,j}(\mu_{kj})-1}\Big)\mathbb{E}\big[OPT^{UB}\big] + \frac{1}{e}\,L(\bar{T},\delta).$$

According to our bridging technique, the above result is an upper bound on the Bayesian regret, which completes the proof. In this upper bound, $\mathbb{E}[OPT^{UB}]$ is the expected value of OPT^{UB} over the prior distribution of the unknown model parameter, and $L(\bar{T},\delta)$ is defined as:

$$\begin{aligned}L(\bar{T},\delta) = \frac{1}{2c_\sigma}\Bigg(&\sqrt{10\,d}\left(\sqrt{\zeta\, T} + \bar{N}_{\max}\right)\log\left(1+\frac{T}{d}\right)\left(\sqrt{d\log\left(1+\frac{T}{d^2}\right)+\log\left(\frac{1}{\delta^2}\right)} + c_\sigma\right)\\ &+ 20\, dD_{\max}\bar{N}_{\max}\left(1+\frac{\bar{N}_{\max}}{d}\right)\log\left(1+\frac{T}{d}\right)\Bigg) + \delta\bar{T},\end{aligned}$$

where both $T = \sum_{m=1}^{M}\bar{N}(m)$ and $\bar{N}_{\max} \;=\; \max\limits_{m\in\mathcal{M}}\bar{N}(m)$ are upper bounded by $\bar{T} - \log(\delta) +$

$\sqrt{(\log(\delta))^2 - 2\bar{T}\log(\delta)}$ and $\left(\bar{T} - \log(\delta) + \sqrt{(\log(\delta))^2 - 2\bar{T}\log(\delta)}\right)/M$ with probability at least $1-\delta$, respectively (see Lemma III.7). □

3.4.4 Discussions of the Main Results

Our PAC algorithm admits a Bayesian regret bound which comes from two major types of loss: (i) the loss associated with contextual learning (batch learning loss), and (ii) the loss associated with the allocation of reusable resources (resource allocation loss).

The batch learning loss is of order $\widetilde{\mathcal{O}}\big(d\sqrt{\bar{T}} + d\,D_{\max}\,\frac{\bar{T}}{M}\big(1+\frac{\bar{T}}{dM}\big)\big)$, where $\bar{T} = \sum_{m=1}^{M}\bar{\lambda}(m)$ and $D_{\max}$ is the maximum number of intervals required for a feedback outcome to be realized. Recall that in the classical online learning problems, the estimator gets updated after sampling one arm. In our setting with *multiple patient arrivals* in each time interval, we update the estimator at the end of each time interval using the feedback outcomes *realized* and the corresponding feature vectors corresponding to the prior batches. For the sake of comparison, $\bar{T} = \sum_{m=1}^{M}\bar{\lambda}(m)$ can be viewed as the number of arrivals over the time horizon in the typical online learning problems. Accordingly, the first term in the batch learning loss matches the minimax lower bound $\Omega(d\sqrt{\bar{T}})$ up to a logarithmic factor provided by [41] for the contextual bandit problems with infinite actions. Recall that the feedback of patients arriving during a time interval may not be realized at the end of each batch. Our theoretical upper bound shows that the *batch learning with delay* impacts the batch learning loss by an additive factor of $d\,D_{\max}\,\frac{\bar{T}}{M}\big(1+\frac{\bar{T}}{dM}\big)$. We highlight that the reward of each type of patient and action pair can be learned separately in a low-dimensional setting. This is a special case of our general batch learning setting in which feature vectors form an orthogonal system. In this case, $d = KJ$ and feature vectors can be constructed as the canonical basis $\{e_n\}_{n=1}^{d}$ of $\mathbb{R}^d$, where e_n is a vector of all zeros except for the n^{th} element which is one. Our upper bound on the batch learning loss still holds for this case.

In contrast with the classical online learning problems, we have *reusable* resources with *limited capacity* in our problem. This results in an extra term in the Bayesian regret called resource allocation loss. This term is upper bounded by $\big(1 - e^{-2\max\limits_{k,j}(\mu_{kj})-1}\big)\mathbb{E}[OPT^{UB}]$. We observe that independent of other parameters, as $\mu_{kj} \to 0$, the upper bound approaches $(1-1/e)\,\mathbb{E}[OPT^{UB}]$. The resource allocation loss partly depends on the blocking loss of the resource allocation mechanism. In particular, the blocking loss is upper bounded by $(1-\alpha)\mathbb{E}[V^{Alg}]$ where $\alpha = \min\limits_{j\in\mathcal{J}}\big\{\sum_{n=0}^{C_j-1}\frac{C_j^n e^{-C_j}}{n!}\big\}$ and V^{Alg} is the total reward obtained by our algorithm without accounting for the reward lost due to blocking. This implies that regardless of the total capacity of each care unit, the blocking loss is upper bounded by $(1-1/e)\mathbb{E}[V^{Alg}]$ because $\alpha \geq 1/e$ by having at least one bed in each care unit.

3.5 Case Study and Empirical Results

Using a hospital system dataset provided by our partner hospital, we evaluate the performance of our PAC algorithm compared to several benchmark policies and a surrogate for the hospital's current policy. In §3.5.1, we first describe our dataset and then calibrate some experimental design choices required for evaluating the empirical performance of our PAC algorithm. In §3.5.2, we provide our empirical results. Finally, some managerial insights are provided in §3.5.3 concerning the impact of implementing our methodology as a decision support tool in practice.

Preliminaries. Our hospital admission control system model, on a high level, needs two interacting mechanisms to (i) adaptively learn the desired patient outcome for different types of patients with respect to different care unit placements, and (ii) judiciously assign patients to different care units with respect to the desired patient outcome. Due to the interest of our partner hospital, we focused on the readmission risk as the main patient outcome. With some effort, our methodology can be extended to capture other possible patient outcomes as well.

In our methodology, the information/feedback revealed for prior care unit placements is used to improve our decision-making by reducing the exposure of patients to less effective decisions and exploring promising ones. In our setting, we should deal with delayed feedback because a non-readmission event cannot be realized until 30 days after the discharge date and a readmission event cannot be realized until the day of readmission. We follow our *on the fly* strategy to deal with delayed feedback. That is, we update the estimator associated with the readmission risk only based on the available information. To account for the limited reusable resources, our algorithm uses a *policy guide* model which captures the effect of lengths of stay on the capacity of care units. This ensures a trade-off between the benefit of better health outcomes by assigning patients to SDUs/ICUs versus the opportunity cost of reserving high-demand beds for potentially complex arriving patients in the future.

3.5.1 Data Description and Experimental Design

Dataset. Our dataset includes more than 10,000 patients from our partner hospital admitted from Emergency Department (ED) or Non-ED. This dataset contains the initial care unit placement for each patient, including ICU, SDU, or GB. It also includes lengths of stay and readmissions to one of the hospitals in the network of hospitals in the area and within the health system. Moreover, our dataset includes the following patient-specific covariates upon admission:

- *Demographics*: age, gender, race, marital status, and insurance type.

- *Diagnosis*: diagnosis upon admission such as renal failure, chronic obstructive pulmonary disease (COPD), sepsis, liver disease, cancer, anemia, myocardial infarction, and hypertension.

- *Risk factors*: patient mortality risk stratum and admission type (ED/Non-ED). Note that the patient mortality risk stratum is a mortality risk measure (MRM) that is available before assigning a patient to a care unit. A mortality risk stratum system has been operational in our partner hospital for multiple years. This measure is an effective summary measure that is not only associated with the risk of death but also correlates with the risk of other adverse health events (see [39] for more information).

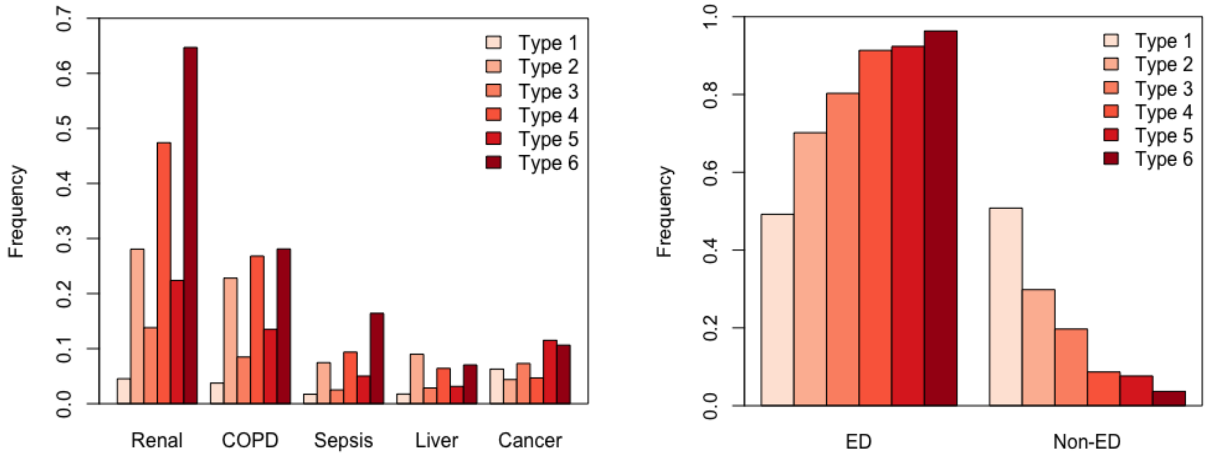
Patient Classification. The first step of our experimental design is to classify patients into several groups with similar characteristics. In our partner hospital, the patient's mortality risk stratum (available upon admission of a patient) is obtained by a predicted severity score based on several different factors, including admission diagnosis (atrial fibrillation, leukemia/lymphoma, metastatic cancer, cancer other than leukemia, lymphoma, cognitive disorder, and neurological conditions), and clinical laboratory results for the tests performed within the preceding 30 days of admission (hemoglobin, platelet count, white blood count, serum troponin, blood urea nitrogen, serum albumin, serum lactate, arterial pH, and arterial partial pressure of oxygen) (see [38]). We adopt a classification approach in which the available patient-specific covariates upon admission are used to cluster patients into 6 different types. We created clusters based on combinations of three ranges for mortality risk and two ranges for the number of diagnoses.

Table 3.1 shows summary statistics with regard to age and gender of each type of patient as well as the mortality risk predicted by our partner hospital. We visualized the frequency of five major diagnoses within each type of patient in Figure 3.2a. Figure 3.2b shows the frequency of admissions from ED and Non-ED for each type of patient. There is a common belief that the care pathways of surgical patients are quite standardized (e.g., [83] and [102]). However, the care pathways of medical patients are more variable, especially for those admitted from the ED. Thus, we put our focus on patients admitted as an ED patient or a non-ED patient to a medical service.

Arrival Process. The frequency of patient arrivals over week days are shown in Figure 3.3. As expected, we see that the frequency of arrivals varies by day and patient type. We assume that the arrival process can be well approximated for each patient type by an independent Poisson distribution for each time interval (day). It should be noted that validating this assumption requires a more complicated analysis which is beyond the scope of this chapter (see [24] and [70] for more information).

Table 3.1: Summary statistics of patient characteristics and average mortality risks by patient types.

Patient type	Age		Female(%)	Mortality(%)
	Mean	SD		
1	52.03	15.56	57.67	0.006
2	53.37	14.11	67.54	0.008
3	65.17	13.49	46.07	0.021
4	66.39	12.46	53.88	0.024
5	74.23	13.92	53.87	0.092
6	75.32	12.28	51.98	0.126



(a) Frequency of five main diagnoses present on admission.

(b) Frequency of admissions from ED and Non-ED.

Figure 3.2: Visualization of the patient characteristics by patient types.

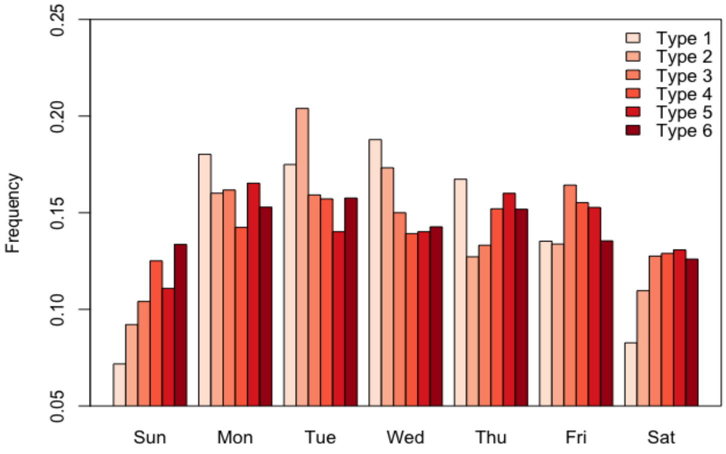
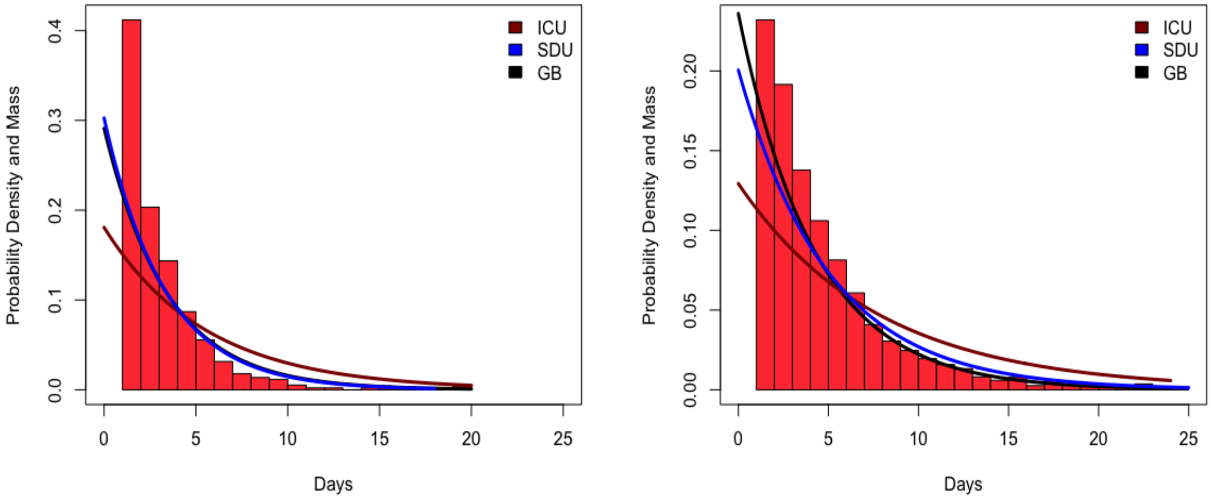


Figure 3.3: Frequency of patient arrivals over week days stratified by each of the patient types.

Length of Stay. Assuming that lengths of stay are following exponential/geometric distributions is a relatively common assumption in the literature (see, e.g., [98], [69], and [27]). Modeling lengths of stay using an exponential/geometric distribution is attractive because (i) it is a good approximation for LOS, and (ii) its memoryless property reduces the complexity of the theoretical analysis of congestion in different care units. We assume that the LOS of each patient type in each care unit follows an exponential distribution. Our analyses shown by Figures 3.4a and 3.4b support this assumption.

Additional Design Considerations. The length of the time intervals and the construction of feature vectors are the two remaining steps of our experimental design. The length of each time interval is considered to be a day for ease of exposition, but it should be thought of as a tuning parameter to be adjusted to suit the application. In general, the smaller the length of intervals, the sooner our algorithm has access to newly realized feedback, and possibly the faster the learning rate. As we consider a low-dimensional setting with six patient types and three care units, we constructed one-hot feature vectors corresponding to all possible combinations of patient types and care units. In a high-dimensional setting, a possible approach to create a feature map is to use a neural network ([73]). In particular, we can train a neural network (either deep or not) on the available historical data to predict the patient outcome. Then, we can obtain a feature map by removing the last layer of the trained neural network.

It is worth noting that real-time evaluation is the ideal way to evaluate our algorithm,



(a) Type 5 patient (the GB line matches the SDU line).

(b) Type 6 patient.

Figure 3.4: Histogram of lengths of stay and fitted distributions of lengths of stay for each care unit.

because counterfactual outcomes are not required when we assign a patient to only one care unit and then obtain the corresponding outcome. Evaluating our algorithm retrospectively based on the observational data is challenging because we need to access counterfactuals in some scenarios. For instance, if our algorithm assigns a patient to the SDU while in the dataset this patient is assigned to the ICU, we do not have access to the readmission outcome of the patient. To address this issue, we estimate the counterfactuals associated with the readmission outcomes using the entire set of observational data. In general, a classification model can be trained (a logistic regression classifier in our case) on the entire data to predict the outcome (whether the patient is readmitted or not) with respect to each care unit. Using this method to estimate counterfactuals may cause potential estimation bias, possibly due to a mismatch between the generative model and the functional form of the readmission outcomes estimator. A more complex voting scheme or a soft ensemble classifier could be a better method for reducing estimation bias and providing accurate estimates (see, e.g., [19] and [26]), but those approaches are beyond the scope of this chapter. Lastly, a limitation of our numerical case study is that care unit placement decisions are more complex than having the single goal of minimizing the readmission rates. In practice, only a portion of the patients can be placed in any of the unit types; however, we did not have access to the data for a sophisticated mode. Our result should be thought of as an upper bound on the potential to reduce readmission rates through care unit placements.

3.5.2 Evaluation and Empirical Results

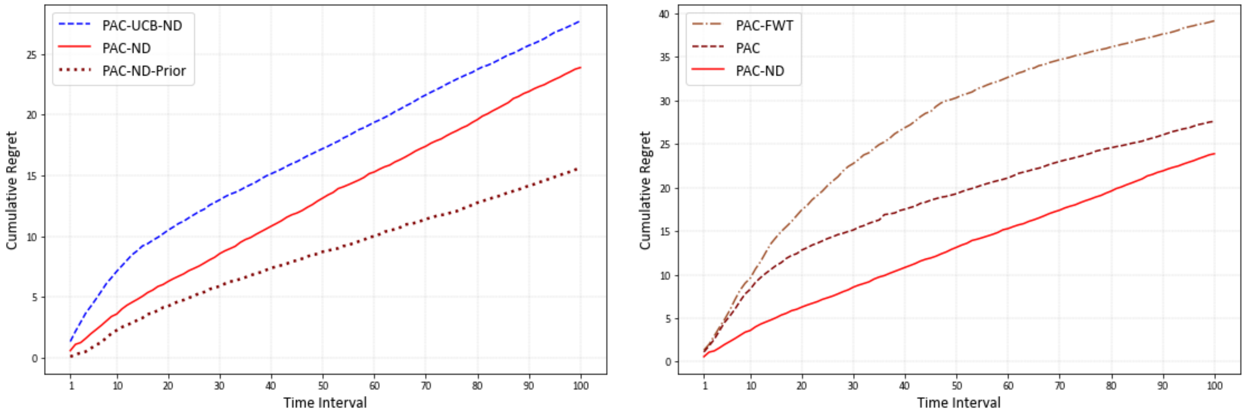
We evaluate the performance of our algorithm with respect to several benchmarks. We consider 10 random permutations of our data in all analyses. We exploit three performance measures, including cumulative regret, cumulative success rate, and distribution of success. The *cumulative regret* measures the difference between the (expected) cumulative reward of the benchmark and our online algorithm, where the benchmark is the upper bound on the optimal expected reward (LP^{UB}). Defining success as the event that a patient is not readmitted, the *cumulative success* measures the total rate of the successful actions upon each time interval. The *distribution of success* describes the distribution of the success rate by the end of the time horizon. We also compare our algorithm with a surrogate that approximates the current policy of our partner hospital.

Cumulative Regret and Success rate. Our algorithm is designed for making care unit placement decisions under limited reusable resources; however, most online learning algorithms in the literature are not capacitated. Thus, we cannot directly compare our algorithm with the existing algorithms in the literature. As a benchmark, we design a variation of PAC, called PAC-UCB, in which the posterior sampling method for learning is replaced by the UCB method ([9], [76]).

First, we compare PAC and PAC-UCB with respect to the cumulative regret. Figure 3.5a illustrates the cumulative regret of each algorithm as a function of the time interval. To exclude the effect of delayed feedback in our comparison, we assumed that patients' feedback is realized *immediately* after making care unit placement decisions. Note that Alg-ND refers to an algorithm using the assumption of no delayed feedback. We considered a non-informative prior for PAC-ND. As can be seen, PAC-ND outperforms PAC-UCB-ND across all time intervals. When there are few samples, the slope of PAC-UCB-ND is higher than the slope of PAC-ND, which may be prohibitively costly in the healthcare setting. The empirical results suggest that a posterior sampling-based algorithm is a better fit for learning in our specific problem.

Good prior information for an algorithm in a Bayesian setting can improve its performance. Our algorithm can admit prior information either from expert opinion or historical data. To illustrate this, we gave PAC-ND access to informative prior information and called it PAC-ND-Prior. We used roughly 30 days to generate prior information for PAC-ND-Prior, and then implemented it on the rest of the data used for other algorithms. Figure 3.5a shows that the magnitude of improvement in the cumulative regret obtained by starting with an informative prior is fairly large. As can be seen, PAC-ND-Prior achieves significantly lower cumulative regret than PAC-ND across all intervals.

Next, we investigate the impact of delayed feedback on regret. Our PAC algorithm uses



(a) No Delay.

(b) With Delay.

Figure 3.5: Cumulative regret of different online algorithms

the *on the fly* strategy to deal with delayed feedback. Our strategy updates the estimator of the unknown vector parameter based on the available information. A simpler but less efficient strategy would be the *fixed waiting time* (FWT) strategy. Note that in our problem the true readmission feedback of a patient will be realized within 30 days after discharge. To deal with the delayed feedback in the FWT strategy, we update the estimator based on the decisions made up to a certain prior time that is long enough to ensure all feedback outcomes are realized. In particular, we update our estimator at interval m using the feedback of decisions made up to interval $m - DL_{\max}$. We set $DL_{\max}$ to 40 which is less than its actual value in our dataset.

Figure 3.5b illustrates the cumulative regret of three variations of our PAC algorithm: (i) PAC, our original algorithm having the on the fly strategy, (ii) PAC-FWT, a variation of PAC in which the FWT strategy is used to deal with delayed feedback, and (iii) PAC-ND, a variation of our PAC algorithm with the assumption of *no delayed feedback*. Note that this is an unrealistic assumption because we assume that a patient's feedback is realized through an oracle immediately after assigning the patient to a care unit. As we expected, PAC-ND outperforms PAC and PAC-FWT since it is using information to which we do not have access in reality. We observe that PAC with the on the fly strategy outperforms PAC-FWT over all time intervals, and it has comparable performance compared to PAC-ND. The reason is the ability of PAC to gain more information using the realized feedback compared to PAC-FWT which uses the same information but with a fixed delay.

Last, we compare PAC and PAC-UCB with respect to the success rate. Figure 3.6 shows the cumulative success rate of PAC and PAC-UCB (both with the on the fly strategy) as a function of time interval, along with the statistical fluctuations (shaded error bars to depict

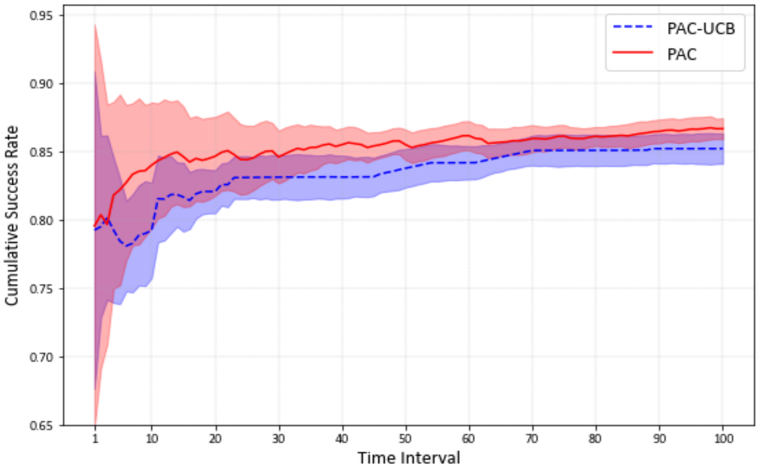


Figure 3.6: Cumulative success rate of PAC and PAC-UCB.

± 1 standard deviation). As can be seen, PAC has a higher average cumulative success rate than PAC-UCB across almost all time intervals. Figure 3.7 illustrates the distribution of the success rate over 100 time intervals. We observe that PAC and PAC-UCB achieve a median success rate of 88.1% and 85.6%, respectively. Overall, the results suggest that PAC performs well compared to PAC-UCB when mean and median matter. The greater performance of PAC with respect to the success rate may provide a significant advantage particularly for healthcare, where any sub-optimal decision for a patient may endanger the patient's health.

Towards Derandomization of PAC – A Deterministic Care Unit Placement Policy. Our PAC algorithm generates a randomized policy for assigning patients to care units. It uses randomization to judiciously utilize the limited capacity in a system with constraints on the expected number of patients in each care unit. Although some randomization in care unit placement is inherent in the current practice, we acknowledge that patients who are told that the system's decisions have some randomness in them may feel uncomfortable. To alleviate this concern, we propose a variation of PAC with a *deterministic* care unit placement strategy called PAC-D.

Recall that in the PAC algorithm, we solve an LP iteratively to find the assignment probabilities, and then we assign a patient of type k to care unit j during interval m following the assignment probability of $x_{kj}(m)$. PAC-D uses the same LP as a policy guide but follows a different strategy to assign patients to care units. In particular, at each interval, it keeps track of the current fraction of patients of type k assigned to care unit j during that interval. Then, the next type k patient is assigned to the care unit j with the highest difference between $x_{kj}(m)$ and the current fraction. We also design a greedy algorithm, called Greedy,

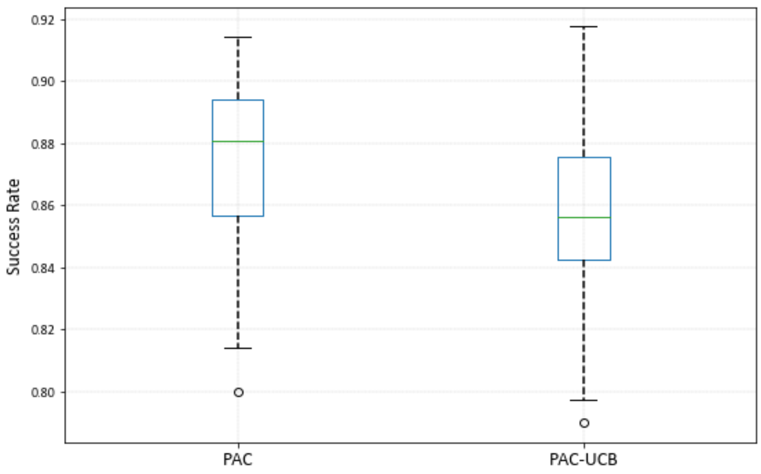


Figure 3.7: Box and whisker plot for the success rate of PAC and PAC-UCB over 100 time intervals.

which does not use the LP but still learns the expected reward. It follows a *greedy* care unit assignment strategy in which a patient is assigned to an available care unit that yields the highest expected reward. In particular, upon arrival of a patient of type k during interval m, Greedy assigns the patient to the care unit j which is $\underset{j \in \mathcal{J}}{\operatorname{argmax}} \, \tilde{r}_{kj}(m)$ subject to real-time capacity constraints.

We compare the success rate of PAC-D with PAC and Greedy. Figure 3.8 shows the distribution of the success rate of all three algorithms over 100 time intervals. We observe that PAC-D achieves a median success rate of 89.8%, which is 1.7% higher than PAC. This improvement may be due to the fact that PAC-D more closely implements the allocations suggested by the LP. It may learn faster over the time horizon because the care unit placements suggested by it are closer to what the LP wants to achieve. The Greedy algorithm achieves a median success rate of 82.0% which is 7.8% and 6.1% less than that of PAC-D and PAC, respectively. The higher performance of PAC-D and PAC over Greedy shows the value of looking ahead when making care unit placement decisions.

Comparison to Hospital's Current Policy. Our partner hospital follows a care unit placement policy that is based on traditional placement criteria and a novel MRM system. A web-based application was developed and implemented in the hospital to generate mortality risk predictions for arriving patients. By the time a patient is ready for bed placement, a prediction of mortality risk is generated and available for clinicians to review. While the real behavior is complicated by additional considerations, the principle is that the MRM system prioritizes high-risk patients with an MRM of 0.2 or higher to be admitted to ICU. Patients with an MRM higher than 0.07 are prioritized to be admitted to SDU, and low-risk

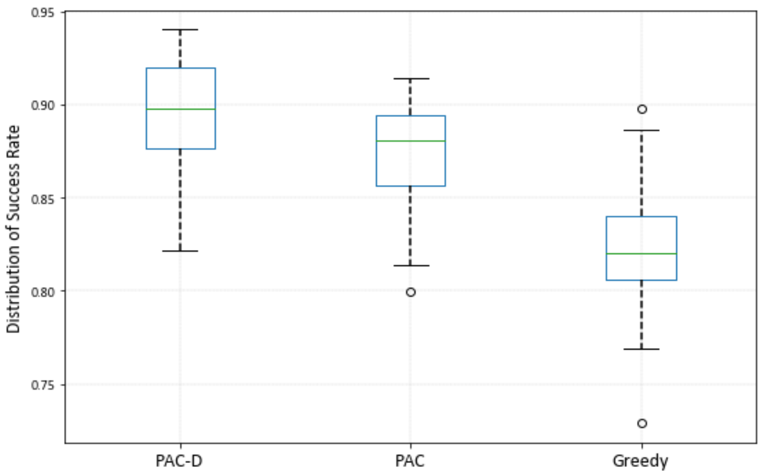


Figure 3.8: Box and whisker plot for the success rate of different online algorithms over 100 time intervals.

patients with an MRM of 0.07 or less are sent to GB. We consider the current policy of the hospital as a benchmark and compare the performance of our algorithm against it. Our comparison indicates the relative change in the readmission rate of a hospital that focuses only on readmission reduction versus a traditional approach augmented by an MRM system.

The difference between PAC and the hospital's policy with respect to the median success rate is 8.6% over 100 intervals, and this difference reaches 10% by considering 30 intervals as a warm-up period for PAC. We also investigated the success rate of PAC and the hospital's policy across different types of patients. Figure 3.9 shows the mean success rate of PAC and the hospital's policy for our six types of patients. As we expected, the mean success rate of PAC is higher for all types of patients except type 5. A closer look at the difference between the success rate of PAC and the hospital's policy reveals that types 2, 4, and 6 have the greatest improvement. According to Figure 3.2a, these three types all have high fractions of the "renal", "COPD", "sepsis", and "liver" disease. Another finding is that the highest improvement in success rate occurs for type 6. According to Table 3.1 and Figure 3.2b, patients of type 6 have the highest mortality risks and a large fraction of them are admitted through ED. Interestingly, type 5 is the only type for which the success rate of PAC is lower than the hospital's policy. The complex trade-off between many factors (e.g., reward, LOS, available capacity at their arrival time) makes it difficult to identify the exact reasons for this. The observed improvements in the readmission outcome of patients, particularly for the complex ones (type 6), seem to be a valuable and interesting advantage of the PAC algorithm.

Our next analysis is to compare PAC and the hospital's policy in terms of daily admission

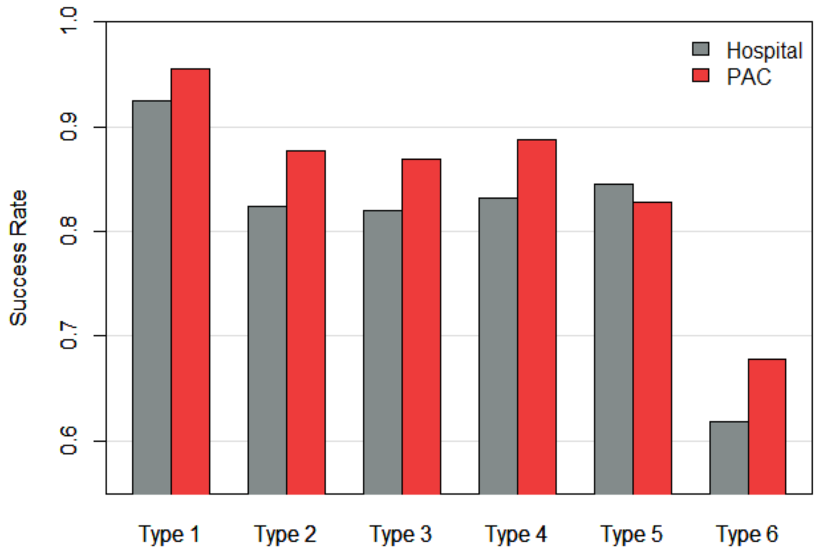
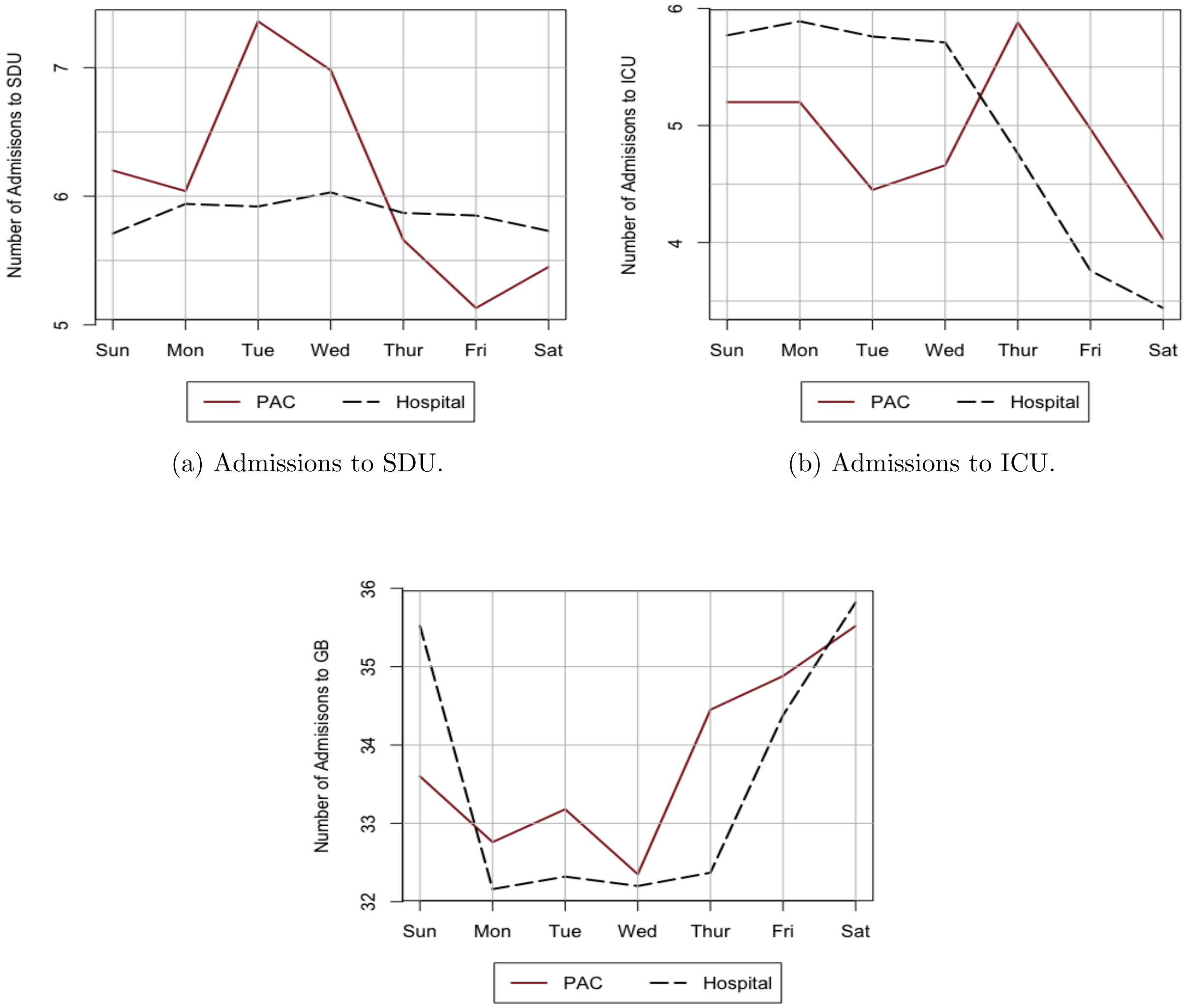


Figure 3.9: Mean success rate for PAC and hospital's current policy across different types of patients.

rates to different care units. To this aim, we simulated the current practice of the hospital based on the available observational data and admission rules of the hospital's MRM system. The average number of admissions by day of the week for SDU, ICU, and GB are illustrated in Figures 3.10a, 3.10b, and 3.10c, respectively. In the hospital's policy, the number of admitted patients to SDU over different days is relatively the same, while more patients were admitted to ICU over the first half of the week compared to its second half. PAC admitted more patients to ICU and fewer patients to SDU during the second half of the week compared to the hospital's policy. The difference between PAC and the hospital's policy in terms of the number of admitted patients to GB is not great over different days. The high success rate of PAC and the observed differences between the number of admissions to different care units suggest that PAC can prioritize patients such that the more complex patients can be assigned to higher levels of care.

3.5.3 Managerial Insights

In this chapter, we considered a fundamental question of how to pursue the goal of readmission reduction through admission control when there is uncertainty regarding the needs of patients. To answer this question, we proposed an optimization-learning approach and designed an online algorithm as an important first step toward answering this question. Next, we considered follow-up questions regarding the efficiency of our approach: Does our optimization-learning approach improve or degrade patient readmission outcomes? What is



(a) Admissions to SDU.

(b) Admissions to ICU.

(c) Admissions to GB.

Figure 3.10: Daily average admissions to GB, SDU, and ICU.

the magnitude of possible improvements?

As discussed before, the costs and benefits associated with different care units are still fundamental questions in the hospital operations management area. One of the main difficulties in answering these questions arises from the fact that such costs and benefits can be highly patient-specific due to the different and often uncertain needs of patients. One key feature of our approach is its ability to provide a time interval- and type-dependent admission policy with respect to readmission risk by leveraging the estimated congestion in the network of units and the future patient arrival pattern. This is done by approximating the dynamics of the system. The other distinctive feature of our approach is its on the fly strategy, which is effective to learn fast when the feedback of prior care unit assignments is delayed in time. We note that it is difficult to make a large impact by merely changing unit placement because a readmission event may depend on many other factors (see, e.g., [59] and [28]). However, our empirical results shown in Figures 3.6-3.10 suggest that our methodology has the potential to be used as the core of a general framework to improve the current policy of hospitals.

Our result shows that the difference between the median success rate of the PAC algorithm and the hospital's policy is up to 10%. According to Figure 3.9, we find that the magnitude of improvements varies across different patient types. That is, the magnitude of improvements in success rates is more significant for type 2, 4, and 6 patients (in increasing order). There are a couple of points to discuss here. Based on Figure 3.2a, we observe that there are two common characteristics among type 2, 4, and 6 patients: (i) a higher fraction of these patients are having a history of comorbidities, especially "renal", "COPD", "sepsis", and "liver" disease, and (ii) a large fraction of them are admitted through ED. Our observations suggest that further attention should be given to these patients to better understand their precise characteristics and confirm that having a history of such diseases is indeed an important factor that should be considered in the care unit placement decisions. Furthermore, our results suggest that greater improvements are likely for patients admitting from ED. One reason could be that the care pathways of Non-ED patients are less variable compared to patients admitting from ED (e.g., [83] and [102]). The high variability in the care pathways of patients admitting from ED can provide more opportunities for our algorithm to leverage patient characteristics.

There might be several underlying reasons for the high success rate of decisions provided by PAC compared to the hospital's policy. Apart from estimating the risk of readmission adaptively, one reason may be that the current policy of the hospital is not able to properly account for the opportunity cost of using each bed type. Our approach accounts for the opportunity cost of using an available bed or saving it for complex patients arrivals in the future. In fact, PAC provides a care unit placement policy based on a patient congestion model. It can be viewed as a time interval- and type-dependent policy which is different from the fixed threshold policy used in the hospital. Our policy can better accommodate the congestion in different care units over time by using arrival rate information, while a fixed threshold policy is hurt by variation in occupancy. Comparing PAC with Greedy in Figure 3.8 supports this claim and highlights the value of looking ahead when making care unit placement decisions.

The results in Figures 3.10a, 3.10b, 3.10c, and the high success rate of unit placement decisions provided by PAC demonstrate that it can judiciously assign patients to different care units. That is, it effectively weighs the overall effects of future arrival rates, LOS, and the relative readmission reduction benefit a patient receives with respect to different care units. The high success rate of PAC for complex patients (i.e., high mortality risk, key commodities, and frequently being admitted through the ED) and its resource utilization, which is roughly consistent with past practice, suggest that PAC can identify and prioritize patients such that the more complex patients are assigned to higher levels of care. It is expected that a higher level of care speeds up the healing process, which can be due to a higher nurse to patient ratio in high-level care units. Our collaborators believe this to be true and literature suggests it as well (see [82]). Thus, we expect our methodology to modestly decrease the average length of stay. Investigating the truth of this hypothesis is beyond the scope of this chapter, but it is worth investigating in future work.

Our methodology can also be useful when an unexpected crisis hits and the health care system must adjust the existing strategies for care unit placements without sufficient data to understand how patients will respond. For this setting, mortality will likely be a more important criterion than readmission, which PAC can accommodate. The Coronavirus disease (COVID-19) pandemic is an example of such an unexpected crisis. To provide proper care to COVID-19 cases and perform infection control, many hospitals reorganized their routine operations and created special care units devoted to COVID-19 patients. While our research study started long before the COVID-19 crisis and we are not considering a special COVID-19 unit, our method can accommodate additional units such as a COVID-19 care unit. Most importantly, the online aspect of our method is well suited the environment with little historical data or undergoing a major change.

3.6 Conclusion

Hospital operations and clinical practice are shifting toward personalization to treat patients better. Hospitals and researchers are eager to learn more about the readmission impact of a care unit placement decision. This is challenging due to the wide variety of patient characteristics, uncertain needs of patients, and the limited number of beds in intensive and intermediate care units. In this study, we proposed an optimization-learning approach for hospital admission control. We introduced a personalized admission control system model, and developed and analyzed a new online algorithm for it. Our algorithm is designed to adaptively learn readmission risks from data through batch learning with delayed feedback and identify the best care unit placements for patients. The aim is to reduce patient readmission rates by capturing the trade-off between the benefit of better health outcomes for patients arriving in the current time interval versus the value of reserving high-demand beds for potentially more complex patients arriving in the future. We analyzed the Bayesian regret of our algorithm and provided a rigorous analytical performance guarantee.

We evaluated the empirical performance of our online algorithm using hospital system data and compared it to several benchmarks and a surrogate for the current policy of our partner hospital. We also investigated the magnitude of improvements in patient readmission outcomes. The case study results showed that our algorithm performs well compared to other benchmarks and reduces the readmission rate up to 10% compared to the current policy of our partner hospital. Our observations suggest that further attention should be given to patients with a history of particular diseases. This helps to better understand the precise characteristics of patients with these diseases and confirm that having a history of such diseases is indeed an important factor that should be considered in the care unit placement decisions. Also, the high success rate of PAC while avoiding over-utilization of SDUs and ICUs suggests that it prioritizes complex patients to be assigned to higher levels of care. We believe our results demonstrate the potential benefits of personalized admission control systems in hospitals.

3.7 Appendix

3.7.1 Appendix A. Proofs of Lemmas 1 and 2

Lemma III.1 (Upper Bound on Optimal Expected Reward). *Let OPT^{UB} be the optimal objective value of LP^{UB} given the model parameter w. Then, the optimal expected reward achieved by the optimal state-dependent policy $\mathbb{E}[V^{\pi^*}|w]$ is upper bounded by OPT^{UB}.*

Proof. Let $x_k^\pi(m) = \{x_{kj}^\pi(m)\}_{j\in\mathcal{J}}$ be the solution of an admissible policy $\pi \in \Pi$. First, we show that $x_k^\pi(m)$ is a feasible solution for LP^{UB}. To do do, we need to show that $x_k^\pi(m)$ satisfies the constraints (3.4)-(3.6) of LP^{UB}. It is easy to see that (3.5) and (3.6) are satisfied by $x_k^\pi(m)$, because policy π induces a distribution over the assignment of patients to care units.

Recall that the flow balance inequality (3.1) holds for each admissible policy π. Since it must be satisfied for any realization, it must also hold after taking expectation on both sides. Then, we have the following:

$$\sum_{k=1}^{K} \bar{\Theta}_{kj}^{\pi}(m) + \sum_{k=1}^{K} \bar{\beta}_{kj}^{\pi}(m) \;\leq\; C_j + \sum_{k=1}^{K} \bar{D}_{kj}^{\pi}(m), \tag{3.12}$$

where $\bar{\Theta}_{kj}^{\pi}(m)$, $\bar{\beta}_{kj}^{\pi}(m)$, and $\bar{D}_{kj}^{\pi}(m)$ are the mean values of $\Theta_{kj}^{\pi}(m)$, $\beta_{kj}^{\pi}(m)$, and $D_{kj}^{\pi}(m)$, respectively.

Assuming that all arrivals occur at the beginning of each interval yields the following upper bound on the expected number of departures of type k patients from care unit j:

$$\bar{D}_{kj}^{\pi}(m) \;\leq\; \big(\bar{\Theta}_{kj}^{\pi}(m) + \bar{\beta}_{kj}^{\pi}(m)\big)\big(1 - e^{-\mu_{kj}}\big), \tag{3.13}$$

where $\mathbb{P}(\text{LOS of a type } k \text{ patient assigned to care unit } j \;\leq\; 1) \;=\; 1 - e^{-\mu_{kj}}$.

By plugging (3.13) into (3.12), we have:

$$\sum_{k=1}^{K} \bar{\beta}_{kj}^{\pi}(m)\, e^{-\mu_{kj}} \;\leq\; C_j - \sum_{k=1}^{K} \bar{\Theta}_{kj}^{\pi}(m)\, e^{-\mu_{kj}}. \tag{3.14}$$

Moreover, since (3.2) must be satisfied for any realization, it must also hold after taking expectation on both sides. Then, we have:

$$\bar{\Theta}_{kj}^{\pi}(m) \;=\; \bar{\Theta}_{kj}^{\pi}(m-1) + \bar{\beta}_{kj}^{\pi}(m-1) - \bar{D}_{kj}^{\pi}(m-1). \tag{3.15}$$

Plugging (3.13) into (3.15) yields the following:

$$\bar{\Theta}_{kj}^{\pi}(m) - \bar{\Theta}_{kj}^{\pi}(m-1)\, e^{-\mu_{kj}} \;\geq\; \bar{\beta}_{kj}^{\pi}(m-1)\, e^{-\mu_{kj}}. \tag{3.16}$$

Now, we are ready to show that $x_k^{\pi}(m)$ satisfies the capacity constraint (3.4) for all $j \in \mathcal{J}$

and $m \in \mathcal{M}$.

$$\begin{aligned}
&\sum_{s=1}^{m} \sum_{k=1}^{K} \lambda_k(s)\; x^{\pi}_{kj}(s)\; \psi_{kj}(s,m) \\
&= \sum_{s=1}^{m} \sum_{k=1}^{K} \psi_{kj}(s,m)\; \bar{\beta}^{\pi}_{kj}(s) \\
&= \sum_{s=1}^{m} \sum_{k=1}^{K} e^{-(m-s+1)\mu_{kj}}\; \bar{\beta}^{\pi}_{kj}(s) \\
&= \sum_{k=1}^{K} e^{-\mu_{kj}}\; \bar{\beta}^{\pi}_{kj}(m) + \sum_{s=1}^{m-1} \sum_{k=1}^{K} e^{-(m-s+1)\mu_{kj}}\; \bar{\beta}^{\pi}_{kj}(s) \\
&\leq C_j - \sum_{k=1}^{K} e^{-\mu_{kj}}\; \bar{\Theta}^{\pi}_{kj}(m) + \sum_{s=1}^{m-1} \sum_{k=1}^{K} e^{-(m-s+1)\mu_{kj}}\; \bar{\beta}^{\pi}_{kj}(s) \\
&\leq C_j - \sum_{k=1}^{K} e^{-\mu_{kj}}\; \bar{\Theta}^{\pi}_{kj}(m) + \sum_{s=1}^{m-1} \sum_{k=1}^{K} e^{-(m-s)\mu_{kj}}\; \left(\bar{\Theta}^{\pi}_{kj}(s+1) - e^{-\mu_{kj}}\; \bar{\Theta}^{\pi}_{kj}(s)\right) \\
&= C_j - \sum_{k=1}^{K} e^{-\mu_{kj}}\; \bar{\Theta}^{\pi}_{kj}(m) + \sum_{k=1}^{K} \left(e^{-\mu_{kj}}\; \bar{\Theta}^{\pi}_{kj}(m) - e^{-m\mu_{kj}}\; \bar{\Theta}^{\pi}_{kj}(1)\right) \\
&= C_j,
\end{aligned}$$

where the first inequality holds because of (3.14), and the second inequality holds because of (3.16). Also, the forth equality follows from the telescoping series, and the last equality holds because the system begins empty and the initial occupancy of any care unit is zero.

Next, it remains to discuss the objective functions. It is easy to observe that the objective function of LP^{UB} is the same as the expected reward of the policy π, i.e., $\mathbb{E}[V^{\pi}|w] = \sum_{m=1}^{M} \sum_{k=1}^{K} \sum_{j=1}^{J} r_{kj}\, \lambda_k(m)\, x^{\pi}_{kj}(m)$. Thus, the optimal expected reward achieved by the optimal state-dependent policy $\mathbb{E}[V^{\pi^*}|w] = \sup_{\pi \in \Pi} \mathbb{E}[V^{\pi}|w]$ is upper bounded by OPT^{UB}, which completes the proof. □

Lemma III.2 (Lower Bound on Optimal Objective Value of LP^{C-UB}). *Let OPT^{C-UB} be the optimal objective value of LP^{C-UB} given the model parameter w. Then, we have $OPT^{C-UB} \geq e^{-2\max_{k,j}(\mu_{kj})}\; OPT^{UB}$.*

Proof. Recall that we defined LP^{C-UB} as follows:

$$\begin{aligned}
\max_{x} \quad & \sum_{m=1}^{M}\sum_{k=1}^{K}\sum_{j=1}^{J} r_{kj}\ \lambda_k(m)\ x_{kj}(m) \\
\text{s.t.} \quad & \sum_{s=1}^{m}\sum_{k=1}^{K}\lambda_k(s)\ x_{kj}(s)\ \psi_{kj}(s,m) \ \ \leq\ \ e^{-2\max\limits_{k,j}(\mu_{kj})}\, C_j, \quad \forall\, j\in\mathcal{J},\ \ \forall\, m\in\mathcal{M} \\
& \sum_{j=1}^{J} x_{kj}(m) \ \ \leq\ \ 1, \quad \forall\, k\in\mathcal{K}, \quad \forall\, m\in\mathcal{M} \\
& x_{kj}(m) \ \ \geq\ \ 0, \quad \forall\ \ k\in\mathcal{K},\ \ \forall\, j\in\mathcal{J},\ \ \forall\, m\in\mathcal{M},
\end{aligned}$$

where $x_{kj}(m)$ is the probability of assigning type k patients to care unit j during interval m. Note that LP^{C-UB} is similar to LP^{UB} and the only difference is that the capacity of care units are scaled down by a multiplier in LP^{C-UB}. Let $x_{kj}(m) = e^{-2\max\limits_{k,j}(\mu_{kj})} x^*_{kj}(m)$, where $x^*_{kj}(m)$ is the optimal solution of LP^{UB}.

First, we show that $e^{-2\max\limits_{k,j}(\mu_{kj})} x^*_{kj}(m)$ is a feasible solution for LP^{C-UB}. Replacing $x_{kj}(m)$ by $e^{-2\max\limits_{k,j}(\mu_{kj})} x^*_{kj}(m)$ for all $k\in\mathcal{K}$, $j\in\mathcal{J}$, and $m\in\mathcal{M}$ in the first constraint of LP^{C-UB}, yields the following:

$$\sum_{s=1}^{m}\sum_{k=1}^{K}\lambda_k(s)\ x^*_{kj}(s)\ \psi_{kj}(s,m) \ \ \leq\ \ C_j, \quad \forall\, j\in\mathcal{J}, \quad \forall\, m\in\mathcal{M}.$$

The above inequality holds since $x^*_{kj}(m)$ is the optimal solution of LP^{UB}. Also, it is obvious that the second constraint of LP^{C-UB} holds as well. Thus, we can conclude that $e^{-2\max\limits_{k,j}(\mu_{kj})} x^*_{kj}(m)$ is a feasible solution for LP^{C-UB}.

Next, we show the the optimal objective value of LP^{C-UB} can be lower bounded as follows, which completes the proof.

$$\begin{aligned}
OPT^{C-UB} \ &\geq\ \sum_{m=1}^{M}\sum_{k=1}^{K}\sum_{j=1}^{J} r_{kj}\ \lambda_k(m)\ x_{kj}(m) \\
&=\ \sum_{m=1}^{M}\sum_{k=1}^{K}\sum_{j=1}^{J} r_{kj}\ \lambda_k(m)\ e^{-2\max\limits_{k,j}(\mu_{kj})} x^*_{kj}(m) \\
&=\ e^{-2\max\limits_{k,j}(\mu_{kj})} \ \sum_{m=1}^{M}\sum_{k=1}^{K}\sum_{j=1}^{J} r_{kj}\ \lambda_k(m)\ x^*_{kj}(m) =\ e^{-2\max\limits_{k,j}(\mu_{kj})}\ OPT^{UB}.
\end{aligned}$$

□

3.7.2 Appendix B. Technical Results on Batch Learning Loss

We provide our technical results for deriving an upper bound on the batch learning loss. In Proposition III.1, we provide a high-probability confidence bound on the expected reward. Proposition III.2 and Lemma III.4 establish upper bounds on two main terms that are essential to calculate the batch learning loss. Proposition III.3 provides a high-probability bound for the batch learning loss.

Proposition III.1 (**Confidence Bound under Batch Learning with Delay**). *For any i, m, and $\delta > 0$, the following holds with probability at least $1 - \delta$.*

$$\left|\sigma\big(\langle\, \phi_{j^*}^{(i)}(m), w\rangle\big) - \sigma\big(\langle\, \phi_{j^*}^{(i)}(m), \hat{w}(m)\rangle\big)\right| \leq \mathrm{Rad}_{j^*}^{(i)}(m),$$

In this upper bound, $\hat{w}(m)$ is the maximum likelihood estimator of the unknown model parameter w at interval m, $\mathrm{Rad}_{j^*}^{(i)}(m) = \frac{1}{4c_\sigma}\left\|\phi_{j^*}^{(i)}(m)\right\|_{U_m^{-1}} \Upsilon_m$, U_m is the design matrix, and

$$\Upsilon_m = \sqrt{d\log\left(1 + \frac{\sum_{s=1}^{m-1}\bar{N}(s)}{\gamma d}\right) + \log\left(\frac{1}{\delta^2}\right)} + 2\sum_{s=\max\{1, m-D_{\max}\}}^{m-1}\ \sum_{i=1}^{\bar{N}(s)}\left\|\phi_{j^*}^{(i)}(s)\right\|_{U_m^{-1}} + c_\sigma,$$

where $\bar{N}(m) = \sum_{k=1}^{K} N_k(m)$, $D_{\max}$ is the maximum number of intervals required for a feedback to be realized, and $c_\sigma = \inf_{w,\phi_{kj}} \dot{\sigma}\big(\langle\, \phi_{kj}, w\rangle\big)$ such that $\dot{\sigma}(\cdot)$ is the derivative of $\sigma(\cdot)$.

Proof. Our aim is to develop a confidence bound which contains the true expected reward with a high probability using the feedback outcomes realized by the end of interval $m-1$ and the available feature vectors upon the end of interval $m-1$.

In our setting, feedback outcomes arrive with delay. Therefore, the information that whether a patient readmits or not is not available immediately after a care unit placement decision is made for the patient. Indeed, the true feedback of a patient may be realized up to a certain number of days after the discharge time. We update the model parameter *on the fly* after each interval. That is, to update the estimator at the end of interval m, we use the information (feature-feedback) of a patient assigned to a care unit in one of the previous intervals if the patient's feedback is *realized* by the end of interval m. Recall that $D^{(i)}(m)$ is the feedback delay of the i^{th} patient who arrived at interval m, which is the number of intervals required for the feedback to be realized. Also, recall that $\mathcal{F}(m) = \big\{(s, i, j^*)\,|\, s + D^{(i)}(s) \leq m - 1;\, i \in \{1, \ldots, \bar{N}(s)\}\big\}$ is the set containing tuples (s, i, j^*) of patients with *realized* readmission feedback outcomes by the end of interval $m-1$. Similarly, we denote $\mathcal{F}^c(m) = \big\{(s, i, j^*)\,|\, s \leq m-1, s + D^{(i)}(s) \geq m;\, i \in \{1, \ldots, \bar{N}(s)\}\big\}$ as the set containing tuples (s, i, j^*) of patients with *unrealized* readmission feedback outcomes by the end of interval $m-1$.

We formally define $\hat{w}(m)$ as the maximum likelihood estimator of $w \in \mathbb{R}^d$ at interval m. The regularized log-likelihood function $\mathcal{G}_m(w)$ can be calculated as:

$$\mathcal{G}_m(w) = \\ \sum_{(s,i,j^*) \in \mathcal{F}(m)} \left[\mathcal{R}^{(i)}_{j^*}(s) \log \sigma\big(\langle\, \phi^{(i)}_{j^*}(s), w \rangle\big) + \Big(1 - \mathcal{R}^{(i)}_{j^*}(s)\Big) \log \Big(1 - \sigma\big(\langle\, \phi^{(i)}_{j^*}(s), w \rangle\big)\Big) \right] - \frac{\kappa}{2} \left\| w \right\|^2 ,$$

where κ is the regularization parameter and $\mathcal{G}_m(w)$ is a strictly concave function of w for $\kappa > 0$.

Next, we need to find the maximum of $\mathcal{G}_m(w)$ in order to obtain the maximum likelihood estimator $\hat{w}(m)$. The gradient of $\mathcal{G}_m(w)$ is obtained as follows:

$$\nabla_w \mathcal{G}_m(w) \\ = \sum_{(s,i,j^*) \in \mathcal{F}(m)} \Big(\mathcal{R}^{(i)}_{j^*}(s) - \sigma\big(\langle\, \phi^{(i)}_{j^*}(s), w \rangle\big) \Big) \phi^{(i)}_{j^*}(s) - \kappa w. \tag{3.17}$$

Thus, $\hat{w}(m)$ is the unique solution of $\nabla_w \mathcal{G}_m(w) = 0$. Moreover, we highlight that the estimator $\hat{w}(m)$ is not updated after each patient and it is only updated after each interval m.

Using the estimator $\hat{w}(m)$, the confidence bound can be derived by the following steps.

First, we define the design matrix U_m and the vector-valued function $h_m(\cdot)$ corresponding to interval m as follows:

$$U_m = \sum_{s=1}^{m-1} \sum_{i=1}^{\bar{N}(s)} \phi^{(i)}_{j^*}(s)\ \phi'^{(i)}_{j^*}(s) + \gamma I; \quad h_m(w) = \sum_{s=1}^{m-1} \sum_{i=1}^{\bar{N}(s)} \sigma\big(\langle\, \phi^{(i)}_{j^*}(s), w \rangle\big)\ \phi^{(i)}_{j^*}(s) + \kappa w,$$

where we set κ to $c_\sigma \gamma > 0$. Note that U_m contains all the feature vectors corresponding to the patients who arrived by the end of interval $m-1$ (see Figure 3.11 for an illustration).

Next, by the *mean value theorem* and the *Lipschitz* property of the logistic function $\sigma(\cdot)$ (see Lemma III.5 in Appendix D), the following holds for any i and m:

$$\left|\sigma\big(\langle\, \phi^{(i)}_{j^*}(m), w \rangle\big) - \sigma\big(\langle\, \phi^{(i)}_{j^*}(m), \hat{w}(m) \rangle\big)\right| \ \le\ \frac{1}{4 c_\sigma} \left\| \phi^{(i)}_{j^*}(m) \right\|_{U_m^{-1}} \left\| h_m(w) - h_m(\hat{w}(m)) \right\|_{U_m^{-1}} . \tag{3.18}$$

According to the definition of $h_m(\cdot)$, we can expand the term $\left\| h_m(\hat{w}(m)) - h_m(w) \right\|_{U_m^{-1}}$

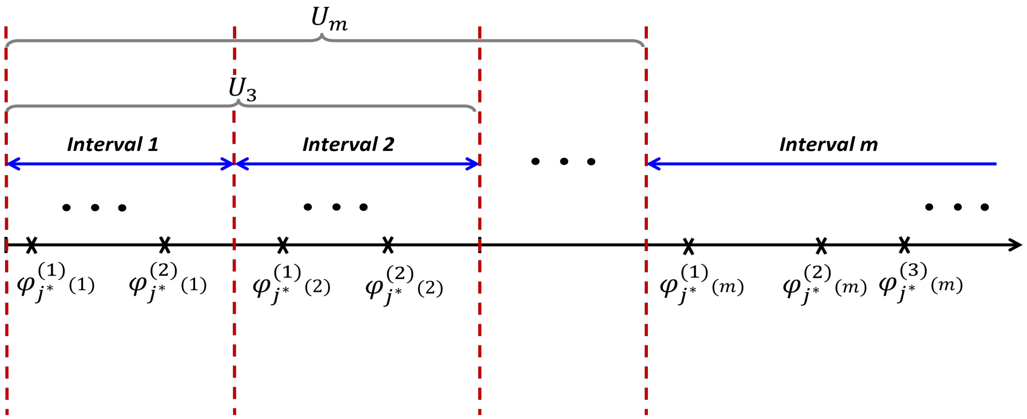


Figure 3.11: The illustration of patient arrivals over different intervals.

on the right-hand side of (3.18) as follows:

$$\begin{aligned}
&\|h_m(\hat{w}(m)) - h_m(w)\|_{U_m^{-1}} \\
&\leq \left\| \sum_{(s,i,j^*)\in\mathcal{F}(m)} \Big(\sigma\big(\langle\, \phi_{j^*}^{(i)}(s), \hat{w}(m)\rangle\big) - \sigma\big(\langle\, \phi_{j^*}^{(i)}(s), w\rangle\big)\Big)\, \phi_{j^*}^{(i)}(s) + \kappa\big(\hat{w}(m) - w\big) \right\|_{U_m^{-1}} \\
&+ \left\| \sum_{(s,i,j^*)\in\mathcal{F}^c(m)} \Big(\sigma\big(\langle\, \phi_{j^*}^{(i)}(s), \hat{w}(m)\rangle\big) - \sigma\big(\langle\, \phi_{j^*}^{(i)}(s), w\rangle\big)\Big)\, \phi_{j^*}^{(i)}(s) \right\|_{U_m^{-1}}, \qquad (3.19)
\end{aligned}$$

where the inequality holds by the triangle inequality.

Recall that $\hat{w}(m)$ is the unique solution of $\nabla_w \mathcal{G}_m(w) = 0$. Then, the following holds by (3.17):

$$\sum_{(s,i,j^*)\in\mathcal{F}(m)} \sigma\big(\langle\, \phi_{j^*}^{(i)}(s), \hat{w}(m)\rangle\big)\, \phi_{j^*}^{(i)}(s) + \kappa\, \hat{w}(m) = \sum_{(s,i,j^*)\in\mathcal{F}(m)} \mathcal{R}_{j^*}^{(i)}(s)\, \phi_{j^*}^{(i)}(s). \qquad (3.20)$$

Accordingly, for the first term on the right-hand side of (3.19), we have:

$$
\begin{aligned}
&\left\| \sum_{(s,i,j^*)\in\mathcal{F}(m)} \Big(\sigma\big(\langle\, \phi_{j^*}^{(i)}(s), \hat{w}(m)\rangle\big) - \sigma\big(\langle\, \phi_{j^*}^{(i)}(s), w\rangle\big)\Big)\, \phi_{j^*}^{(i)}(s) + \kappa\big(\hat{w}(m) - w\big) \right\|_{U_m^{-1}} \\
&= \left\| \sum_{(s,i,j^*)\in\mathcal{F}(m)} \Big(\mathcal{R}_{j^*}^{(i)}(s) - \sigma\big(\langle\, \phi_{j^*}^{(i)}(s), w\rangle\big)\Big)\, \phi_{j^*}^{(i)}(s) - \kappa w \right\|_{U_m^{-1}} \\
&\le \left\| \sum_{(s,i,j^*)\in\mathcal{F}(m)} \xi_{j^*}^{(i)}(s)\ \phi_{j^*}^{(i)}(s) \right\|_{U_m^{-1}} + \kappa\, \|w\|_{U_m^{-1}} \\
&\le \left\| \sum_{s=1}^{m-1} \sum_{i=1}^{\bar{N}(s)} \xi_{j^*}^{(i)}(s)\ \phi_{j^*}^{(i)}(s) \right\|_{U_m^{-1}} + \left\| \sum_{(s,i,j^*)\in\mathcal{F}^c(m)} \xi_{j^*}^{(i)}(s)\ \phi_{j^*}^{(i)}(s) \right\|_{U_m^{-1}} + c_\sigma,
\end{aligned}
$$

where the first equality holds by (3.20). The first inequality holds by the triangle inequality and having $\xi_{j^*}^{(i)}(m) = \mathcal{R}_{j^*}^{(i)}(m) - \sigma(\langle\, \phi_{j^*}^{(i)}(m), w\rangle)$. The last inequality holds by the triangle inequality and having $\|w\|_{U_m^{-1}}^2 \le \rho_{\min}^{-1}(U_m)\, \|w\|^2 \le \gamma^{-1}\, \|w\|^2 \le \gamma^{-1}$.

Plugging the above inequality in (3.19), we have:

$$
\begin{aligned}
\|h_m(\hat{w}(m)) - h_m(w)\|_{U_m^{-1}} \le\ & \underbrace{\left\| \sum_{s=1}^{m-1} \sum_{i=1}^{\bar{N}(s)} \xi_{j^*}^{(i)}(s)\ \phi_{j^*}^{(i)}(s) \right\|_{U_m^{-1}} + c_\sigma}_{\text{Term I}} \\
&+ \underbrace{\left\| \sum_{(s,i,j^*)\in\mathcal{F}^c(m)} \Big(\sigma\big(\langle\, \phi_{j^*}^{(i)}(s), \hat{w}(m)\rangle\big) - \sigma\big(\langle\, \phi_{j^*}^{(i)}(s), w\rangle\big)\Big)\, \phi_{j^*}^{(i)}(s) \right\|_{U_m^{-1}}}_{\text{Term II}} \\
&+ \underbrace{\left\| \sum_{(s,i,j^*)\in\mathcal{F}^c(m)} \xi_{j^*}^{(i)}(s)\ \phi_{j^*}^{(i)}(s) \right\|_{U_m^{-1}}}_{\text{Term III}}.
\end{aligned}
$$

In the following, we bound each term.

Term I: Let $\mathcal{H}_m^0$ be a sigma algebra generated by the feature vectors and the noise values of the patients who arrived by the end of interval $m-1$. Note that $\xi_{j^*}^{(i)}(m)$ can be viewed as a 1-sub-Gaussian random variable and the sequence $\{\sum_{s=1}^{m-1} \sum_{i=1}^{\bar{N}(s)} \xi_{j^*}^{(i)}(s)\, \phi_{j^*}^{(i)}(s)\}_{m\in\mathcal{M}}$ is a *martingale* adapted to $\{\mathcal{H}_m^0\}_{m\in\mathcal{M}}$. This martingale can be bounded with a high probability (see Theorem 1 in [1]). Accordingly, for any i, m, and $\delta > 0$, the following holds with probability at least $1-\delta$:

$$\left\|\sum_{s=1}^{m-1}\sum_{i=1}^{\bar{N}(s)} \xi_{j^*}^{(i)}(s)\ \phi_{j^*}^{(i)}(s)\right\|_{U_m^{-1}} \leq \sqrt{2\log\left(\frac{\det(U_m)^{1/2}\det(\gamma I)^{-1/2}}{\delta}\right)}.$$

Note that U_m is a positive definite matrix, $\mathrm{tr}(U_m)$ is equal to the summation of its eigenvalues, and $\det(U_m)$ is equal to the product of its eigenvalues. Therefore, by $\mathrm{tr}(U_m) \leq \mathrm{tr}(\gamma I)+\sum_{s=1}^{m-1}\bar{N}(s)$ and using the inequality of arithmetic and geometric, we have $\det(U_m) \leq \left(\frac{1}{d}\ \mathrm{tr}(U_m)\right)^d \leq \left(\gamma + \frac{\sum_{s=1}^{m-1}\bar{N}(s)}{d}\right)^d$. Thus, the root-squared term in the above inequality can be further simplified as:

$$\begin{aligned}
2\log\left(\frac{\det(U_m)^{1/2}\det(\gamma I)^{-1/2}}{\delta}\right) &= 2\log\left(\det(U_m)^{1/2}\right) + 2\log\left(\frac{\det(\gamma I)^{-1/2}}{\delta}\right)\\
&\leq d\log\left(\gamma + \frac{\sum_{s=1}^{m-1}\bar{N}(s)}{d}\right) + 2\log\left(\det(\gamma I)^{-1/2}\right) + 2\log\left(\frac{1}{\delta}\right)\\
&= d\log\left(\gamma + \frac{\sum_{s=1}^{m-1}\bar{N}(s)}{d}\right) + d\log\left(\frac{1}{\gamma}\right) + 2\log\left(\frac{1}{\delta}\right)\\
&= d\log\left(1 + \frac{\sum_{s=1}^{m-1}\bar{N}(s)}{\gamma d}\right) + \log\left(\frac{1}{\delta^2}\right).
\end{aligned}$$

<u>Terms II & III</u>: First, we bound Term II. Recall that $\sigma(\cdot)$ is the logistic function. The absolute value of the difference of two logistic functions is upper bounded by one. Then, we have the following by the triangle inequality:

$$\begin{aligned}
&\left\|\sum_{(s,i,j^*)\in\mathcal{F}^c(m)}\left(\sigma\left(\langle\,\phi_{j^*}^{(i)}(s),\hat{w}(m)\rangle\right) - \sigma\left(\langle\,\phi_{j^*}^{(i)}(s),w\rangle\right)\right)\ \phi_{j^*}^{(i)}(s)\right\|_{U_m^{-1}}\\
&\leq \sum_{(s,i,j^*)\in\mathcal{F}^c(m)}\left\|\left(\sigma\left(\langle\,\phi_{j^*}^{(i)}(s),\hat{w}(m)\rangle\right) - \sigma\left(\langle\,\phi_{j^*}^{(i)}(s),w\rangle\right)\right)\ \phi_{j^*}^{(i)}(s)\right\|_{U_m^{-1}}\\
&\leq \sum_{(s,i,j^*)\in\mathcal{F}^c(m)}\left\|\phi_{j^*}^{(i)}(s)\right\|_{U_m^{-1}}.
\end{aligned}$$

Recall that $D_{\max}$ is the maximum number of intervals required for a feedback to be realized. Then, the feedback outcomes of all patients who arrived before interval $m-D_{\max}$ are realized by the end of interval $m-1$. However, we do not know whether the feedback outcomes of other patients, who arrived on interval $m-D_{\max}$ or subsequent intervals including interval $m-1$, are realized or not by the end of interval $m-1$. Then, it is not hard to see that the number of such patients or the cardinality of set $\mathcal{F}^c(m)$ can be upper bounded by $\sum_{s=\max\{1,m-D_{\max}\}}^{m-1} \bar{N}(s)$ where $\bar{N}(s) = \sum_{k=1}^{K} N_k(s)$. Then, we have:

$$\sum_{(s,i,j^*)\in\mathcal{F}^c(m)} \left\| \phi_{j^*}^{(i)}(s) \right\|_{U_m^{-1}} \le \sum_{s=\max\{1,m-D_{\max}\}}^{m-1} \sum_{i=1}^{\bar{N}(s)} \left\| \phi_{j^*}^{(i)}(s) \right\|_{U_m^{-1}}.$$

Noting that the absolute value of the noise is upper bounded by one, Term III can be similarly upper bounded as follows:

$$\left\| \sum_{(s,i,j^*)\in\mathcal{F}^c(m)} \xi_{j^*}^{(i)}(s)\ \phi_{j^*}^{(i)}(s) \right\|_{U_m^{-1}} \le \sum_{s=\max\{1,m-D_{\max}\}}^{m-1} \sum_{i=1}^{\bar{N}(s)} \left\| \phi_{j^*}^{(i)}(s) \right\|_{U_m^{-1}}.$$

Putting the results derived for three terms together, we have:

$$\| h_m(\hat{w}(m)) - h_m(w) \|_{U_m^{-1}} \le \Upsilon_m, \tag{3.21}$$

where Υ_m is defined as:

$$\sqrt{d \log\left(1 + \frac{\sum_{s=1}^{m-1} \bar{N}(s)}{\gamma d}\right) + \log\left(\frac{1}{\delta^2}\right)} + 2 \sum_{s=\max\{1,m-D_{\max}\}}^{m-1} \sum_{i=1}^{\bar{N}(s)} \left\| \phi_{j^*}^{(i)}(s) \right\|_{U_m^{-1}} + c_\sigma.$$

Finally, the proof is completed by (3.18) and (3.21).

$$\left| \sigma\big(\langle\, \phi_{j^*}^{(i)}(m), w\rangle\big) - \sigma\big(\langle\, \phi_{j^*}^{(i)}(m), \hat{w}(m)\rangle\big) \right| \le \mathrm{Rad}_{j^*}^{(i)}(m),$$

where $\mathrm{Rad}_{j^*}^{(i)}(m) = \frac{1}{4c_\sigma} \left\| \phi_{j^*}^{(i)}(m) \right\|_{U_m^{-1}} \Upsilon_m$. □

Lemma III.3 (Bound on Nested Summation of Feature Vectors). *Let* $\{\phi_{j^*}^{(i)}(m)\}_{m\in\mathcal{M},i\in\bar{N}(m)}$ *be a sequence of feature vectors in* $\mathbb{R}^d$*. Then, the following holds almost surely:*

$$\begin{aligned}&\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\left\|\phi_{j^*}^{(i)}(m)\right\|_{U_m^{-1}}\left(\sum_{s=\max\{1,m-D_{\max}\}}^{m-1}\sum_{n=1}^{\bar{N}(s)}\left\|\phi_{j^*}^{(n)}(s)\right\|_{U_m^{-1}}\right)\\&\leq 10\,dD_{\max}\bar{N}_{\max}\left(1+\frac{\bar{N}_{\max}}{\gamma}\right)\log\left(\frac{\gamma+T}{\gamma}\right).\end{aligned}$$

Proof. The proof consists of two main steps.
Step 1. Using $ab\leq(a^2+b^2)/2$ for $a,b\in\mathbb{R}$, we have:

$$\begin{aligned}&\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\left\|\phi_{j^*}^{(i)}(m)\right\|_{U_m^{-1}}\left(\sum_{s=\max\{1,m-D_{\max}\}}^{m-1}\sum_{n=1}^{\bar{N}(s)}\left\|\phi_{j^*}^{(n)}(s)\right\|_{U_m^{-1}}\right)\\&\leq\frac{1}{2}\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\sum_{s=\max\{1,m-D_{\max}\}}^{m-1}\sum_{n=1}^{\bar{N}(s)}\left(\left\|\phi_{j^*}^{(i)}(m)\right\|_{U_m^{-1}}^2+\left\|\phi_{j^*}^{(n)}(s)\right\|_{U_m^{-1}}^2\right).\end{aligned}$$

Next, by the Sherman-Morrison formula, we have $\left\|\phi_{j^*}^{(n)}(s)\right\|_{U_m^{-1}}\leq\left\|\phi_{j^*}^{(n)}(s)\right\|_{U_s^{-1}}$ for all $s\leq m$. Accordingly, the following holds:

$$\begin{aligned}&\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\sum_{s=\max\{1,m-D_{\max}\}}^{m-1}\sum_{n=1}^{\bar{N}(s)}\left(\left\|\phi_{j^*}^{(i)}(m)\right\|_{U_m^{-1}}^2+\left\|\phi_{j^*}^{(n)}(s)\right\|_{U_m^{-1}}^2\right)\\&\leq\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\sum_{s=\max\{1,m-D_{\max}\}}^{m-1}\sum_{n=1}^{\bar{N}(s)}\left(\left\|\phi_{j^*}^{(i)}(m)\right\|_{U_m^{-1}}^2+\left\|\phi_{j^*}^{(n)}(s)\right\|_{U_s^{-1}}^2\right)\\&\leq\bar{N}_{\max}D_{\max}\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\left\|\phi_{j^*}^{(i)}(m)\right\|_{U_m^{-1}}^2+\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\sum_{s=\max\{1,m-D_{\max}\}}^{m-1}\sum_{n=1}^{\bar{N}(s)}\left\|\phi_{j^*}^{(n)}(s)\right\|_{U_s^{-1}}^2\\&\leq 2\,\bar{N}_{\max}D_{\max}\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\left\|\phi_{j^*}^{(i)}(m)\right\|_{U_m^{-1}}^2,\end{aligned}$$

where the last inequality holds because $\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\sum_{s=\max\{1,m-D_{\max}\}}^{m-1}\sum_{n=1}^{\bar{N}(s)}\left\|\phi_{j^*}^{(n)}(s)\right\|_{U_s^{-1}}^2\leq\bar{N}_{\max}D_{\max}\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\left\|\phi_{j^*}^{(i)}(m)\right\|_{U_m^{-1}}^2$.
Step 2. In the second step, we upper bound the summation of the weighted norms of feature vectors.

Let $E_m = \sum_{i=1}^{\bar{N}(m)} \phi_{j^*}^{(i)}(m)\, \phi_{j^*}^{'(i)}(m)$. Then, we have:

$$\sum_{m=1}^{M} \sum_{i=1}^{\bar{N}(m)} \left\| \phi_{j^*}^{(i)}(m) \right\|_{U_m^{-1}}^2 = \sum_{m=1}^{M} \operatorname{tr}\left(U_m^{-1} E_m\right).$$

Let $\nu_{m,j}$ for all $j \in \{1, \dots, d\}$ be the eigenvalues of U_m. By Lemma 11 of [8] and using a recursion technique, we have:

$$\sum_{m=1}^{M} \operatorname{tr}\left(U_m^{-1} E_m\right) \leq 10 \sum_{m=1}^{M} \sum_{j=1}^{d} \frac{\nu_{m+1,j} - \nu_{m,j}}{\nu_{m,j}},$$

where we have $\nu_{m+1,j} \geq \nu_{m,j}$ and $\nu_{1,j} = \gamma$ for all m and j.

To upper bound the above term, we establish the following decomposition:

$$\begin{aligned} &\sum_{m=1}^{M} \operatorname{tr}\left(U_m^{-1} E_m\right) \\ &\leq 10 \sum_{m=1}^{M} \sum_{j=1}^{d} \frac{\nu_{m+1,j} - \nu_{m,j}}{\nu_{m+1,j}} + 10 \left(\sum_{m=1}^{M} \sum_{j=1}^{d} \frac{\nu_{m+1,j} - \nu_{m,j}}{\nu_{m,j}} - \sum_{m=1}^{M} \sum_{j=1}^{d} \frac{\nu_{m+1,j} - \nu_{m,j}}{\nu_{m+1,j}} \right). \end{aligned}$$

Similar to the technique used in Lemma 3 of [58], we bound the first term as follows:

$$\sum_{m=1}^{M} \sum_{j=1}^{d} \frac{\nu_{m+1,j} - \nu_{m,j}}{\nu_{m+1,j}} \leq \sum_{j=1}^{d} \int\limits_{\nu_{1,j}}^{\nu_{M+1,j}} \frac{1}{t}\, dt = \sum_{j=1}^{d} \log\left(\frac{\nu_{M+1,j}}{\nu_{1,j}}\right) \leq d \log\left(\frac{\gamma + T}{\gamma}\right),$$

where the last inequality holds since for any $z \in \mathbb{R}^d$, we have:

$$\begin{aligned} z'\, U_{M+1}\, z &= z' \left(\sum_{m=1}^{M} \sum_{i=1}^{\bar{N}(m)} \phi_{j^*}^{(i)}(m)\, \phi_{j^*}^{'(i)}(m) + \gamma I \right) z \\ &= \gamma \left\| z \right\|^2 + \sum_{m=1}^{M} \sum_{i=1}^{\bar{N}(m)} \langle \phi_{j^*}^{(i)}(m), z \rangle^2 \leq \gamma \left\| z \right\|^2 + T \left\| z \right\|^2. \end{aligned}$$

This implies that $\nu_{M+1,j} \leq \gamma + T$ for all $j \in \{1, \dots, d\}$.

Next, we upper bound the second term. By a simple algebra, we have:

$$\sum_{m=1}^{M} \sum_{j=1}^{d} \frac{\nu_{m+1,j} - \nu_{m,j}}{\nu_{m,j}} - \sum_{m=1}^{M} \sum_{j=1}^{d} \frac{\nu_{m+1,j} - \nu_{m,j}}{\nu_{m+1,j}} = \sum_{m=1}^{M} \sum_{j=1}^{d} \frac{\left(\nu_{m+1,j} - \nu_{m,j}\right)^2}{\nu_{m,j}\, \nu_{m+1,j}}.$$

Note that $\nu_{m,j} \geq \nu_{1,j} = \gamma$. Also, $\operatorname{tr}(U_{m+1}) = \operatorname{tr}(U_m) + \operatorname{tr}(E_m)$, then we have:

$$\nu_{m+1,j} - \nu_{m,j} \leq \operatorname{tr}(E_m) = \sum_{i=1}^{\bar{N}(m)} \left\| \phi_{j^*}^{(i)}(m) \right\|^2 \leq \bar{N}_{\max}.$$

Accordingly, the second term can be upper bounded as follows:

$$\sum_{m=1}^{M} \sum_{j=1}^{d} \frac{\left(\nu_{m+1,j} - \nu_{m,j}\right)^2}{\nu_{m,j}\, \nu_{m+1,j}} \leq \frac{\bar{N}_{\max}}{\gamma} \sum_{m=1}^{M} \sum_{j=1}^{d} \frac{\nu_{m+1,j} - \nu_{m,j}}{\nu_{m+1,j}} \leq \frac{d\, \bar{N}_{\max}}{\gamma} \log\left(\frac{\gamma + T}{\gamma}\right).$$

Thus, we have:

$$\begin{aligned}
&\sum_{m=1}^{M} \operatorname{tr}\left(U_m^{-1} E_m\right) \\
&\leq 10 \sum_{m=1}^{M} \sum_{j=1}^{d} \frac{\nu_{m+1,j} - \nu_{m,j}}{\nu_{m+1,j}} + 10 \left(\sum_{m=1}^{M} \sum_{j=1}^{d} \frac{\nu_{m+1,j} - \nu_{m,j}}{\nu_{m,j}} - \sum_{m=1}^{M} \sum_{j=1}^{d} \frac{\nu_{m+1,j} - \nu_{m,j}}{\nu_{m+1,j}} \right) \\
&\leq 10\, d \left(1 + \frac{\bar{N}_{\max}}{\gamma}\right) \log\left(\frac{\gamma + T}{\gamma}\right).
\end{aligned}$$

Finally, putting the results obtained by the two steps together completes the proof. □

Now, we state Proposition III.2 and Lemma III.4, which establish high-probability bounds on two main terms that are essential for calculating the batch learning loss in the next step (see Proposition III.3).

Proposition III.2 (Bound on Difference between Upper and Lower Bounds of Expected Reward). *For any $\delta > 0$, the following holds with probability at least $1 - \delta$.*

$$\begin{aligned}
&\mathbb{E}\left[\sum_{m=1}^{M} \sum_{i=1}^{\bar{N}(m)} \sum_{j=1}^{J} \left(UB_j^{(i)}(m)\, x_j^{(i),Alg^*}(m) - LB_j^{(i)}(m)\, x_j^{(i),Alg^*}(m) \right) \right] \\
&\leq \frac{1}{2c_\sigma} \Bigg(\sqrt{10\, d} \left(\sqrt{\zeta\, T} + \bar{N}_{\max} \right) \log\left(1 + \frac{T}{d}\right) \left(\sqrt{d \log\left(1 + \frac{T}{d^2}\right) + \log\left(\frac{1}{\delta^2}\right)} + c_\sigma \right) \\
&\qquad + 20\, d D_{\max} \bar{N}_{\max} \left(1 + \frac{\bar{N}_{\max}}{d}\right) \log\left(1 + \frac{T}{d}\right) \Bigg),
\end{aligned}$$

where $UB_j^{(i)}(m)$ and $LB_j^{(i)}(m)$ are the largest and smallest possible estimated values for the expected reward of the i^{th} patient who arrived at interval m and was assigned to care unit j, respectively. Also, $\bar{N}_{\max}$ and T can be upper bounded with a high probability (see Lemma

III.7).

Proof. Let $UB_j^{(i)}(m)$ and $LB_j^{(i)}(m)$ be the sequences of real-valued functions which can be defined as:

$$UB_j^{(i)}(m) = \min\left\{1, \max_{\tau\in\Gamma_m} \sigma\big(\langle\, \phi_j^{(i)}(m), \tau\rangle\big)\right\}; \quad LB_j^{(i)}(m) = \max\left\{0, \min_{\tau\in\Gamma_m} \sigma\big(\langle\, \phi_j^{(i)}(m), \tau\rangle\big)\right\},$$

where set Γ_m can be defined as follows:

$$\Gamma_m = \Big\{w \in \mathbb{R}^d \Big| \, \|h_m(\hat{w}(m)) - h_m(w)\|_{U_m^{-1}} \le \Upsilon_m\Big\}. \tag{3.22}$$

Since we cannot assign a patient to more than one care unit, we have:

$$\begin{aligned}
&\mathbb{E}\left[\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\sum_{j=1}^{J}\Big(UB_j^{(i)}(m)\; x_j^{(i),Alg^*}(m) - LB_j^{(i)}(m)\; x_j^{(i),Alg^*}(m)\Big)\right] \\
&= \mathbb{E}\left[\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\sum_{j=1}^{J}\Big(UB_j^{(i)}(m) - LB_j^{(i)}(m)\Big)\, \mathbb{1}\{j^{(i),Alg^*}(m) = j\}\right] \\
&= \mathbb{E}\left[\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\Big(UB_{j^*}^{(i)}(m) - LB_{j^*}^{(i)}(m)\Big)\right].
\end{aligned}$$

By Proposition III.1, we established the following bound for any i, m, and $\delta > 0$, which holds with probability at least $1 - \delta$:

$$\big|\sigma\big(\langle\, \phi_{j^*}^{(i)}(m), w\rangle\big) - \sigma\big(\langle\, \phi_{j^*}^{(i)}(m), \hat{w}(m)\rangle\big)\big| \le \mathrm{Rad}_{j^*}^{(i)}(m).$$

Next, we define $\widetilde{UB}_j^{(i)}(m)$ and $\widetilde{LB}_j^{(i)}(m)$ as follows:

$$\begin{aligned}
\widetilde{UB}_j^{(i)}(m) &= \sigma\big(\langle\, \phi_j^{(i)}(m), \hat{w}(m)\rangle\big) + \mathrm{Rad}_j^{(i)}(m), \\
\widetilde{LB}_j^{(i)}(m) &= \sigma\big(\langle\, \phi_j^{(i)}(m), \hat{w}(m)\rangle\big) - \mathrm{Rad}_j^{(i)}(m).
\end{aligned}$$

Since $UB_{j^*}^{(i)}(m) \le \widetilde{UB}_{j^*}^{(i)}(m)$ and $LB_{j^*}^{(i)}(m) \ge \widetilde{LB}_{j^*}^{(i)}(m)$, we have:

$$UB_{j^*}^{(i)}(m) - LB_{j^*}^{(i)}(m) \;\le\; \widetilde{UB}_{j^*}^{(i)}(m) - \widetilde{LB}_{j^*}^{(i)}(m) \le 2\,\mathrm{Rad}_{j^*}^{(i)}(m).$$

Summing over all patients and taking expectation on both sides of the above inequality yield the following:

$$\begin{aligned}&\mathbb{E}\left[\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\left(UB_{j^*}^{(i)}(m)-LB_{j^*}^{(i)}(m)\right)\right]\\&\leq 2\,\mathbb{E}\left[\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\mathrm{Rad}_{j^*}^{(i)}(m)\right]=\mathbb{E}\left[\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\frac{1}{2c_\sigma}\left\|\phi_{j^*}^{(i)}(m)\right\|_{U_m^{-1}}\Upsilon_m\right],\end{aligned}\tag{3.23}$$

where,

$$\Upsilon_m=\sqrt{d\log\left(1+\frac{\sum_{s=1}^{m-1}\bar{N}(s)}{\gamma d}\right)+\log\left(\frac{1}{\delta^2}\right)}+2\sum_{s=\max\{1,m-D_{\max}\}}^{m-1}\sum_{n=1}^{\bar{N}(s)}\left\|\phi_{j^*}^{(n)}(s)\right\|_{U_m^{-1}}+c_\sigma.$$

By Lemma III.6 (Appendix D), the following holds when $\gamma=d$:

$$\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\left\|\phi_{j^*}^{(i)}(m)\right\|_{U_m^{-1}}\leq\sqrt{10\,d}\left(\sqrt{M\bar{N}_{\max}}+\bar{N}_{\max}\right)\log\left(1+\frac{T}{d}\right).$$

Next, by Lemma III.3, the following holds when $\gamma=d$:

$$\begin{aligned}&\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\left\|\phi_{j^*}^{(i)}(m)\right\|_{U_m^{-1}}\left(\sum_{s=\max\{1,m-D_{\max}\}}^{m-1}\sum_{n=1}^{\bar{N}(s)}\left\|\phi_{j^*}^{(n)}(s)\right\|_{U_m^{-1}}\right)\\&\leq 10\,dD_{\max}\bar{N}_{\max}\left(1+\frac{\bar{N}_{\max}}{d}\right)\log\left(1+\frac{T}{d}\right).\end{aligned}$$

Finally, the proof is completed by replacing the above results into (3.23).

$$\begin{aligned}&\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\frac{1}{2c_\sigma}\left\|\phi_{j^*}^{(i)}(m)\right\|_{U_m^{-1}}\Upsilon_m\leq\\&\frac{1}{2c_\sigma}\left(\sqrt{10\,d}\left(\sqrt{\zeta\,T}+\bar{N}_{\max}\right)\log\left(1+\frac{T}{d}\right)\left(\sqrt{d\log\left(1+\frac{T}{d^2}\right)+\log\left(\frac{1}{\delta^2}\right)}+c_\sigma\right)\right.\\&\qquad\left.+20\,dD_{\max}\bar{N}_{\max}\left(1+\frac{\bar{N}_{\max}}{d}\right)\log\left(1+\frac{T}{d}\right)\right),\end{aligned}$$

where $\bar{N}_{\max}\leq\zeta\,\frac{\sum_{m=1}^{M}\bar{N}(m)}{M}$. □

Lemma III.4 (Bound on Difference between Expected Reward and its Upper Bound). *For any $\delta > 0$, the following bound holds with probability at least $1 - \delta$.*

$$\mathbb{E}\left[\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\sum_{j=1}^{J}\Big(r_j^{(i)}(m)\, x_j^{(i),Aux^*}(m) - UB_j^{(i)}(m)\, x_j^{(i),Aux^*}(m)\Big)\right] \le \delta\bar{T}.$$

Proof. Since we cannot assign a patient to more than one care unit, we have:

$$\begin{aligned}
&\mathbb{E}\left[\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\sum_{j=1}^{J}\Big(r_j^{(i)}(m)\, x_j^{(i),Aux^*}(m) - UB_j^{(i)}(m)\, x_j^{(i),Aux^*}(m)\Big)\right]\\
&= \mathbb{E}\left[\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\sum_{j=1}^{J}\Big(r_j^{(i)}(m) - UB_j^{(i)}(m)\Big)\,\mathbb{1}\{j^{(i),Aux^*}(m) = j\}\right]\\
&= \mathbb{E}\left[\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\Big(r_{j^*}^{(i)}(m) - UB_{j^*}^{(i)}(m)\Big)\right].
\end{aligned}$$

Note that we have $r_j^{(i)}(m) \in (0,1)$ and $UB_j^{(i)}(m) \ge 0$ for any i, j, and m. Thus, the following holds:

$$r_{j^*}^{(i)}(m) - UB_{j^*}^{(i)}(m) \le \mathbb{1}\big\{r_{j^*}^{(i)}(m) > UB_{j^*}^{(i)}(m)\big\}.$$

By taking expectation on both sides of the above inequality, we have:

$$\mathbb{E}\big[r_{j^*}^{(i)}(m) - UB_{j^*}^{(i)}(m)\big] \le \mathbb{P}\big(r_{j^*}^{(i)}(m) > UB_{j^*}^{(i)}(m)\big).$$

By Proposition III.1, the following holds for any i, m, and $\delta > 0$ with probability at least $1 - \delta$:

$$\big|\sigma\big(\langle\, \phi_{j^*}^{(i)}(m), w\rangle\big) - \sigma\big(\langle\, \phi_{j^*}^{(i)}(m), \hat{w}(m)\rangle\big)\big| \le \mathrm{Rad}_{j^*}^{(i)}(m).$$

This confidence bound implies that:

$$\mathbb{P}\big(\widetilde{LB}_{j^*}^{(i)}(m) \le r_{j^*}^{(i)}(m) \le \widetilde{UB}_{j^*}^{(i)}(m)\big) \;\ge\; 1 - \delta. \tag{3.24}$$

Then, we have:

$$\mathbb{E}\big[r_{j^*}^{(i)}(m) - UB_{j^*}^{(i)}(m)\big] \le \mathbb{P}\big(r_{j^*}^{(i)}(m) > UB_{j^*}^{(i)}(m)\big) \;=\; \mathbb{P}\big(r_{j^*}^{(i)}(m) > \widetilde{UB}_{j^*}^{(i)}(m)\big) \le \delta.$$

Note that if $UB_{j^*}^{(i)}(m) < 1$, we have $UB_{j^*}^{(i)}(m) = \widetilde{UB}_{j^*}^{(i)}(m)$ by definition. The equality holds because when $r_{j^*}^{(i)}(m) > UB_{j^*}^{(i)}(m)$, we have $UB_{j^*}^{(i)}(m) < 1$ which implies that $UB_{j^*}^{(i)}(m) = \widetilde{UB}_{j^*}^{(i)}(m)$. The last inequality holds by (3.24).

Finally, the proof is completed by the following:

$$\mathbb{E}\left[\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\left(r_{j^*}^{(i)}(m) - UB_{j^*}^{(i)}(m)\right)\right] \leq \delta\sum_{m=1}^{M}\bar{\lambda}(m) = \delta\,\bar{T}.$$

□

Next, we provide a high-probability bound on the difference between the true expected rewards obtained by the care unit placement decisions made by the auxiliary algorithm and the PAC algorithm. This term is $\mathbb{E}\big[V^{Aux} - V^{Alg}\big]$ which measures the batch learning loss.

Proposition III.3 (Batch Learning Loss). *For any $\delta > 0$, the following holds with probability at least $1-\delta$.*

$$\begin{aligned}&\sum_{m=1}^{M}\sum_{k=1}^{K}\sum_{j=1}^{J}\mathbb{E}\left[r_{kj}\,\lambda_k(m)\,x_{kj}^{Aux^*}(m) - r_{kj}\,\lambda_k(m)\,x_{kj}^{Alg^*}(m)\right]\\ &\leq \frac{1}{2c_\sigma}\Bigg(\sqrt{10\,d}\left(\sqrt{\zeta\,T} + \bar{N}_{\max}\right)\log\left(1+\frac{T}{d}\right)\left(\sqrt{d\log\left(1+\frac{T}{d^2}\right)+\log\left(\frac{1}{\delta^2}\right)}+c_\sigma\right)\\ &\qquad + 20\,dD_{\max}\bar{N}_{\max}\left(1+\frac{\bar{N}_{\max}}{d}\right)\log\left(1+\frac{T}{d}\right)\Bigg) + \delta\bar{T},\end{aligned}$$

where $\bar{N}_{\max}$ and T can be upper bounded with a high probability (see Lemma III.7).

Proof. In the PAC algorithm, we assume a prior distribution over the unknown model parameter w and we update the posterior distribution as new information is received after each interval. At the beginning of each interval, we take a random sample $\tilde{w}(m)$ from the posterior distribution to estimate the expected rewards and solve an LP to update the assignment probabilities. Recall that $x_k^{Alg^*}(m) = \{x_{kj}^{Alg^*}(m)\}_{j\in\mathcal{J}}$ and $x_k^{Aux^*}(m) = \{x_{kj}^{Aux^*}(m)\}_{j\in\mathcal{J}}$ are the optimal solutions of the PAC algorithm and the auxiliary algorithm, respectively. Let $\mathcal{H}_m$ be the history available by the end of interval $m-1$, which can be defined as:

$$\begin{aligned}\mathcal{H}_m = \Big\{\big(\varphi^{(i)}(s), j^{(i)}(s), x_{kj}(s)\big)\big| s \leq m-1,\ \forall\, k\in\mathcal{K},\ \forall\, j\in\mathcal{J},\ i\in\{1,\ldots,\bar{N}(s)\}\Big\}\cup\\ \Big\{\mathcal{R}_{j^*}^{(i)}(s)\big| s + D^{(i)}(s) \leq m-1,\ i\in\{1,\ldots,\bar{N}(s)\}\Big\},\end{aligned}$$

where $\varphi^{(i)}(s)$ is the context vector of the i^{th} patient who arrived at interval s.

Note that the assignment probabilities for interval m are updated at the beginning of the interval using the estimations obtained by the history $\mathcal{H}_m$. To upper bound the batch learning loss, we need to argue that $x_k^{Alg^*}(m)$ and $x_k^{Aux^*}(m)$ are *identically distributed* conditional on history $\mathcal{H}_m$, that is, $\mathbb{P}(x_k^{Alg^*}(m)|\mathcal{H}_m) = \mathbb{P}(x_k^{Aux^*}(m)|\mathcal{H}_m)$. First, in the PAC algorithm, assigning a patient of type k who arrived at interval m to a care unit depends on the expected reward under $\tilde{w}(m)$ (i.e., $\tilde{r}_{kj}(m)$) and the assignment probabilities for intervals $\{1, \ldots, m-1\}$ obtained by the PAC algorithm. Second, because $\tilde{w}(m)$ is sampled from the posterior distribution $\mathbb{P}(w|\mathcal{H}_m)$, vector parameters w and $\tilde{w}(m)$ are *identically distributed* conditional on history $\mathcal{H}_m$, that is, $\mathbb{P}(\tilde{w}(m)|\mathcal{H}_m) = \mathbb{P}(w|\mathcal{H}_m)$. Also, conditional on history $\mathcal{H}_m$, the assignment probabilities for intervals $\{1, \ldots, m-1\}$ are the same for both the PAC and auxiliary algorithms. Thus, we conclude that $\mathbb{P}(x_k^{Alg^*}(m)|\mathcal{H}_m) = \mathbb{P}(x_k^{Aux^*}(m)|\mathcal{H}_m)$.

Accordingly, we can derive the following decomposition:

$$\begin{aligned}
&\sum_{m=1}^{M}\sum_{k=1}^{K}\sum_{j=1}^{J}\mathbb{E}\big[r_{kj}\,\lambda_k(m)\,x_{kj}^{Aux^*}(m) - r_{kj}\,\lambda_k(m)\,x_{kj}^{Alg^*}(m)\big] \\
&= \sum_{m=1}^{M}\sum_{k=1}^{K}\sum_{j=1}^{J}\mathbb{E}\left[\mathbb{E}\big[r_{kj}\,\lambda_k(m)\,x_{kj}^{Aux^*}(m) - r_{kj}\,\lambda_k(m)\,x_{kj}^{Alg^*}(m)\big|\mathcal{H}_m\big]\right] \\
&= \sum_{m=1}^{M}\sum_{k=1}^{K}\sum_{j=1}^{J}\mathbb{E}\Big[\mathbb{E}\big[UB_{kj}(m)\,\lambda_k(m)\,x_{kj}^{Alg^*}(m) - r_{kj}\,\lambda_k(m)\,x_{kj}^{Alg^*}(m)\big|\mathcal{H}_m\big]\Big] \\
&+ \sum_{m=1}^{M}\sum_{k=1}^{K}\sum_{j=1}^{J}\mathbb{E}\Big[\mathbb{E}\big[r_{kj}\,\lambda_k(m)\,x_{kj}^{Aux^*}(m) - UB_{kj}(m)\,\lambda_k(m)\,x_{kj}^{Aux^*}(m)\big|\mathcal{H}_m\big]\Big] \\
&= \mathbb{E}\left[\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\sum_{j=1}^{J}\Big(UB_j^{(i)}(m)\;x_j^{(i),Alg^*}(m) - r_j^{(i)}(m)\;x_j^{(i),Alg^*}(m)\Big)\right] \\
&+ \mathbb{E}\left[\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\sum_{j=1}^{J}\Big(r_j^{(i)}(m)\;x_j^{(i),Aux^*}(m) - UB_j^{(i)}(m)\;x_j^{(i),Aux^*}(m)\Big)\right],
\end{aligned} \tag{3.25}$$

where $UB_{kj}(m) = \min\left\{1, \max_{\tau\in\Gamma_m}\sigma\big(\langle\,\phi_{kj}, \tau\rangle\big)\right\}$. The first equality holds by the law of iterated expectation. The second equality holds because $\mathbb{P}(x_k^{Alg^*}(m)|\mathcal{H}_m) = \mathbb{P}(x_k^{Aux^*}(m)|\mathcal{H}_m)$ and $UB_{kj}(m)$ is a deterministic function given the history $\mathcal{H}_m$. The above decomposition let us leverage the connection between the posterior sampling-based algorithms and UCB algorithms ([95]).

By Proposition III.1, the following confidence bound holds with probability at least $1-\delta$:

$$\left|\sigma\big(\langle\, \phi_{j^*}^{(i)}(m), w\rangle\big) - \sigma\big(\langle\, \phi_{j^*}^{(i)}(m), \hat{w}(m)\rangle\big)\right| \le \mathrm{Rad}_{j^*}^{(i)}(m).$$

Accordingly, when the above confidence bound holds, the two terms in (**??**) can be upper bounded using Proposition III.2 and Lemma III.4.

Recall that we defined $UB_j^{(i)}(m)$ and $LB_j^{(i)}(m)$ as sequences of real-valued functions of $\mathcal{H}_m$ and feature vector $\phi_j^{(i)}(m)$:

$$UB_j^{(i)}(m) = \min\left\{1, \max_{\tau\in\Gamma_m} \sigma\big(\langle\, \phi_j^{(i)}(m), \tau\rangle\big)\right\}; \quad LB_j^{(i)}(m) = \max\left\{0, \min_{\tau\in\Gamma_m} \sigma\big(\langle\, \phi_j^{(i)}(m), \tau\rangle\big)\right\}.$$

Using the above definitions, we have the following for the first term in (**??**):

$$\begin{aligned}
&\mathbb{E}\left[\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\sum_{j=1}^{J}\Big(UB_j^{(i)}(m)\, x_j^{(i),Alg^*}(m) - r_j^{(i)}(m)\, x_j^{(i),Alg^*}(m)\Big)\right]\\
&\le \mathbb{E}\left[\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\sum_{j=1}^{J}\Big(UB_j^{(i)}(m)\, x_j^{(i),Alg^*}(m) - LB_j^{(i)}(m)\, x_j^{(i),Alg^*}(m)\Big)\right].
\end{aligned}$$

By Proposition III.2, we have:

$$\begin{aligned}
&\mathbb{E}\left[\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\sum_{j=1}^{J}\Big(UB_j^{(i)}(m)\, x_j^{(i),Alg^*}(m) - LB_j^{(i)}(m)\, x_j^{(i),Alg^*}(m)\Big)\right]\\
&\le \frac{1}{2c_\sigma}\Bigg(\sqrt{10\, d}\,\Big(\sqrt{\zeta\, T} + \bar{N}_{\max}\Big)\log\left(1+\frac{T}{d}\right)\left(\sqrt{d\log\left(1+\frac{T}{d^2}\right)+\log\left(\frac{1}{\delta^2}\right)}+c_\sigma\right)\\
&\qquad + 20\, dD_{\max}\bar{N}_{\max}\left(1+\frac{\bar{N}_{\max}}{d}\right)\log\left(1+\frac{T}{d}\right)\Bigg).
\end{aligned}$$

By Lemma III.4, the second term in (**??**) can be upper bounded as:

$$\mathbb{E}\left[\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\sum_{j=1}^{J}\Big(r_j^{(i)}(m)\, x_j^{(i),Aux^*}(m) - UB_j^{(i)}(m)\, x_j^{(i),Aux^*}(m)\Big)\right] \le \delta\bar{T}.$$

Finally, putting the last two results together completes the proof. □

3.7.3 Appendix C. Technical Result on Blocking Loss

Recall that the PAC algorithm provides care unit placement decisions by following a policy guide that takes into account the average lengths of stay and generates state-independent care unit assignment probabilities. Therefore, a patient assigned to a care unit may get *blocked* because the selected care unit is being utilized at capacity. Accordingly, the algorithm may incur a certain *loss* because of the possibility of blocking. In this section, we derive an upper bound on this blocking loss. Note that we assume zero rewards for the blocked patients. In practice, they can be sent to GB where hospitals often have sufficient beds.

Proposition III.4 (Blocking Loss). *Given the model parameter w, let V^{Alg} denote the total reward obtained by the PAC algorithm excluding the loss due to blocking, also let V^{BL} denote the loss due to blocking. Then, the following holds:*

$$\mathbb{E}[V^{BL}|w] \leq \left(1 - \min_{j\in\mathcal{J}}\left\{\sum_{n=0}^{C_j-1}\frac{C_j^n e^{-C_j}}{n!}\right\}\right)\mathbb{E}[V^{Alg}|w].$$

Proof. Recall that for an admissible policy π, the average number of type k patients who arrived to care unit j during interval m can be calculated by $\bar{\beta}^{\pi}_{kj}(m) = \lambda_k(m)\, x^{\pi}_{kj}(m)$. However, this does not hold for the PAC algorithm because some patients assigned to a care unit by the algorithm may get blocked. Thus, we should make a distinction between the number of patients assigned to a care unit and the number of patients successfully accepted to the care unit. We define $Z_{kj}(m)$ as the number of type k patients assigned to care unit j, and we denote its mean value by $\bar{Z}_{kj}(m) = \lambda_k(m)\, x^{Alg^*}_{kj}(m)$. We also define $\beta_{kj}(m)$ as the number of type k patients successfully accepted to care unit j, and we denote its mean value by $\bar{\beta}_{kj}(m)$, where $\bar{\beta}_{kj}(m) \leq \bar{Z}_{kj}(m)$.

For our analysis, we need to implicitly model $\beta_{kj}(m)$ which is a function of $\Theta_{kj}(m)$ and $Z_{kj}(m)$. The current $\Theta_{kj}(m)$ is a complex function of the number of arrivals in the prior intervals and the number of available beds in the corresponding care unit. Due to this complexity, we work with an upper bound on $\Theta_{kj}(m)$ with a better structure. To do so, similar to [88], we define a *relaxed system* for which we assume (i) arriving patients are not subjected to limited capacity C_j, and (ii) patient arrivals occur at the end of the intervals, which implies that patients cannot depart in the same interval they arrive. We define $\Theta^{(\mathrm{R})}_{kj}(m)$ as the number of type k patients in care unit j at the beginning of interval m in the relaxed system, where $\Theta^{(\mathrm{R})}_{kj}(m) \geq \Theta_{kj}(m)$. Note that $Z_{kj}(m)$ is following a Poisson distribution with mean $\bar{Z}_{kj}(m) = \lambda_k(m)\, x^{Alg^*}_{kj}(m)$. Then, the number of admitted patients from each prior interval is also following a Poisson distribution that has been thinned by the probability of patients' departure. Therefore, it is easy to see that $\Theta^{(\mathrm{R})}_{kj}(m)$ is also following a Poisson distribution with the following mean:

$$\begin{aligned}\bar{\Theta}^{(\mathrm{R})}_{kj}(m) &= \sum_{s=1}^{m-1} e^{-(m-s-1)\mu_{kj}}\ \bar{Z}_{kj}(s) \\ &= e^{2\mu_{kj}} \sum_{s=1}^{m-1} e^{-(m-s+1)\mu_{kj}}\ \bar{Z}_{kj}(s) \\ &= e^{2\mu_{kj}} \left(\sum_{s=1}^{m} \psi_{kj}(s,m)\ \bar{Z}_{kj}(s) - e^{-\mu_{kj}}\ \bar{Z}_{kj}(m) \right). \qquad (3.26)\end{aligned}$$

Let $\beta^{(\mathrm{R})}_{kj}(m)$ denote the number of type k patients who arrived to care unit j during interval m and were successfully accepted into the care unit when we use $\Theta^{(\mathrm{R})}_{kj}(m)$ rather than $\Theta_{kj}(m)$ in the *main system*. We also define $Z_j(m)$ with mean $\bar{Z}_j(m) = \sum_{k=1}^{K} \bar{Z}_{kj}(m)$, and $\Theta^{(\mathrm{R})}_{j}(m)$ with mean $\bar{\Theta}^{(\mathrm{R})}_{j}(m) = \sum_{k=1}^{K} \bar{\Theta}^{(\mathrm{R})}_{kj}(m)$. Now, we are ready to derive an upper bound on the actual expected (total) reward of the algorithm.

$$\begin{aligned}&\mathbb{E}[V^{Alg} - V^{BL} | w] \\ &= \mathbb{E}\left[\sum_{m=1}^{M}\sum_{k=1}^{K}\sum_{j=1}^{J} r_{kj}\, \beta_{kj}(m) |\, w \right] \\ &\geq \mathbb{E}\left[\sum_{m=1}^{M}\sum_{k=1}^{K}\sum_{j=1}^{J} r_{kj}\, \beta^{(\mathrm{R})}_{kj}(m) |\, w \right] \\ &\geq \sum_{m=1}^{M}\sum_{j=1}^{J}\sum_{n=0}^{C_j-1}\sum_{u=0}^{\infty} \mathbb{E}\left[\textstyle\sum_{k=1}^{K} r_{kj}\, \beta^{(\mathrm{R})}_{kj}(m) \mid w,\ \Theta^{(\mathrm{R})}_{j}(m) = n, Z_j(m) = u \right] \mathbb{P}(\Theta^{(\mathrm{R})}_{j}(m) = n)\ \mathbb{P}(Z_j(m) = u) \\ &\geq \sum_{m=1}^{M}\sum_{j=1}^{J}\sum_{n=0}^{C_j-1}\sum_{u=0}^{C_j-n} \mathbb{E}\left[\textstyle\sum_{k=1}^{K} r_{kj}\, \beta^{(\mathrm{R})}_{kj}(m) \mid w,\ \Theta^{(\mathrm{R})}_{j}(m) = n, Z_j(m) = u \right] \mathbb{P}(\Theta^{(\mathrm{R})}_{j}(m) = n)\ \mathbb{P}(Z_j(m) = u) \\ &= \sum_{m=1}^{M}\sum_{j=1}^{J}\sum_{n=0}^{C_j-1}\sum_{u=0}^{C_j-n} \mathbb{E}\left[\textstyle\sum_{k=1}^{K} r_{kj}\, Z_{kj}(m) \mid w,\ \Theta^{(\mathrm{R})}_{j}(m) = n, Z_j(m) = u \right] \mathbb{P}(\Theta^{(\mathrm{R})}_{j}(m) = n)\ \mathbb{P}(Z_j(m) = u),\end{aligned}$$

$$(3.27)$$

where the first inequality holds since $\beta^{(\mathrm{R})}_{kj}(m) \leq \beta_{kj}(m)$. The second inequality holds by the law of total probability and removing all terms in which the number of patients in a care unit at the beginning of an interval is equal or greater than the total capacity of the care unit. In the third inequality, we removed all terms in which the number of arrivals is more than the available number of beds in a care unit at the beginning of an interval. Note that in the the last equality, we replaced $\beta_{kj}^{(\mathrm{R})}(m)$ by $Z_{kj}(m)$ since there is no rejection when the number of arrivals is equal or less then the available number of beds.

Let $\breve{r}_j(m) = \frac{\sum_{k=1}^{K} r_{kj}\,\bar{Z}_{kj}(m)}{\sum_{k=1}^{K} \bar{Z}_{kj}(m)}$. By multiplying $\breve{r}_j(m)$ by $Z_j(m) = \sum_{k=1}^{K} Z_{kj}(m)$ and taking expectation, we obtain $\mathbb{E}\left[\breve{r}_j(m)\,Z_j(m) \mid w\right] = \mathbb{E}\left[\sum_{k=1}^{K} r_{kj}\,Z_{kj}(m) \mid w\right]$. Accordingly, we have the following by (3.27):

$$
\begin{aligned}
&\mathbb{E}[V^{Alg} - V^{BL}|w] \\
&\geq \sum_{m=1}^{M}\sum_{j=1}^{J}\sum_{n=0}^{C_j-1}\sum_{u=0}^{C_j-n} \mathbb{E}\left[\sum_{k=1}^{K} r_{kj}\,Z_{kj}(m) \mid \Theta_j^{(\mathrm{R})}(m)=n, Z_j(m)=u\right] \mathbb{P}(\Theta_j^{(\mathrm{R})}(m)=n)\,\mathbb{P}(Z_j(m)=u) \\
&= \sum_{m=1}^{M}\sum_{j=1}^{J}\sum_{n=0}^{C_j-1}\sum_{u=0}^{C_j-n} \breve{r}_j(m)\,u\,\mathbb{P}(\Theta_j^{(\mathrm{R})}(m)=n)\,\mathbb{P}(Z_j(m)=u) \\
&= \sum_{m=1}^{M}\sum_{j=1}^{J}\sum_{n=0}^{C_j-1}\sum_{u=0}^{C_j-n} \breve{r}_j(m)\,u\,\left(\frac{\left(\bar{\Theta}_j^{(\mathrm{R})}(m)\right)^n e^{-\bar{\Theta}_j^{(\mathrm{R})}(m)}}{n!}\right)\left(\frac{\left(\bar{Z}_j(m)\right)^u e^{-\bar{Z}_j(m)}}{u!}\right). \qquad (3.28)
\end{aligned}
$$

Next, we show that $\bar{Z}_j(m) + \bar{\Theta}_j^{(\mathrm{R})}(m)$ can be upper bounded for all $j \in \mathcal{J}$.

$$
\begin{aligned}
\bar{Z}_j(m) + \bar{\Theta}_j^{(\mathrm{R})}(m) &= \sum_{k=1}^{K}\bar{Z}_{kj}(m) + \sum_{k=1}^{K} e^{2\mu_{kj}}\left(\sum_{s=1}^{m}\psi_{kj}(s,m)\,\bar{Z}_{kj}(s) - e^{-\mu_{kj}}\,\bar{Z}_{kj}(m)\right) \\
&= \sum_{k=1}^{K} e^{2\mu_{kj}}\left(\sum_{s=1}^{m}\psi_{kj}(s,m)\,\bar{Z}_{kj}(s) - e^{-\mu_{kj}}\,\bar{Z}_{kj}(m) + e^{-2\mu_{kj}}\,\bar{Z}_{kj}(m)\right) \\
&\leq \sum_{s=1}^{m}\sum_{k=1}^{K} e^{2\mu_{kj}}\,\psi_{kj}(s,m)\,\bar{Z}_{kj}(s) \\
&\leq e^{2\max\limits_{k,j}(\mu_{kj})}\sum_{s=1}^{m}\sum_{k=1}^{K}\,\psi_{kj}(s,m)\,\bar{Z}_{kj}(s) \\
&\leq\; C_j,
\end{aligned}
$$

where the first equality holds by (3.26). The first inequality holds since $-e^{-\mu_{kj}}\,\bar{Z}_{kj}(m) + e^{-2\mu_{kj}}\bar{Z}_{kj}(m)$ is less than zero. The last inequality holds by having $\bar{Z}_{kj}(m) = \lambda_k(m)\,x_{kj}^{Alg^*}(m)$ and the capacity constraint.

According to the above inequality and (3.28), the proof is completed by the following:

$$\begin{aligned}
&\mathbb{E}[V^{Alg} - V^{BL}|w] \\
&\geq \sum_{m=1}^{M}\sum_{j=1}^{J}\sum_{n=0}^{C_j-1}\sum_{u=0}^{C_j-n} \breve{r}_j(m)\, u \left(\frac{\left(C_j - \bar{Z}_j(m)\right)^n e^{-(C_j-\bar{Z}_j(m))}}{n!}\right)\left(\frac{\left(\bar{Z}_j(m)\right)^u e^{-\bar{Z}_j(m)}}{u!}\right) \\
&\geq \sum_{m=1}^{M}\sum_{j=1}^{J}\left(\breve{r}_j(m)\, \bar{Z}_j(m) \sum_{n=0}^{C_j-1}\frac{C_j^n e^{-C_j}}{n!}\right) \\
&\geq \min_{j\in\mathcal{J}}\left\{\sum_{n=0}^{C_j-1}\frac{C_j^n e^{-C_j}}{n!}\right\}\sum_{m=1}^{M}\sum_{j=1}^{J}\breve{r}_j(m)\,\bar{Z}_j(m) \\
&= \min_{j\in\mathcal{J}}\left\{\sum_{n=0}^{C_j-1}\frac{C_j^n e^{-C_j}}{n!}\right\}\mathbb{E}[V^{Alg}|w],
\end{aligned}$$

where the second inequality follows by a simple algebra. The equality holds because we have:

$$\sum_{m=1}^{M}\sum_{j=1}^{J}\breve{r}_j(m)\,\bar{Z}_j(m) = \sum_{m=1}^{M}\sum_{k=1}^{K}\sum_{j=1}^{J} r_{kj}\,\bar{Z}_{kj}(m) = \mathbb{E}[V^{Alg}|w].$$

3.7.4 Appendix D. Standard Known Results

We provide some known results from the existing literature. For completeness, we provide self-contained and more expository versions of the original proofs.

Lemma III.5 (Initial Confidence Bound on Expected Reward). *For any i and m, the following holds almost surely:*

$$\left|\sigma\left(\langle\, \phi_{j^*}^{(i)}(m), w\rangle\right) - \sigma\left(\langle\, \phi_{j^*}^{(i)}(m), \hat{w}(m)\rangle\right)\right| \leq \frac{1}{4c_\sigma}\left\|\phi_{j^*}^{(i)}(m)\right\|_{U_m^{-1}} \left\|h_m(w) - h_m(\hat{w}(m))\right\|_{U_m^{-1}}.$$

Proof. We adapt the proof proposed by [50] for our regularized setting. Let $w^0 = \epsilon\, w + (1-\epsilon)\, \hat{w}(m)$ with $0 < \epsilon < 1$. According to the mean value theorem for vector-valued functions, we have:

$$h_m(w) - h_m(\hat{w}(m)) = \left(\int_0^1 \mathrm{J}_{h_m(w^0)}\, d\epsilon\right)(w - \hat{w}(m)),$$

where Jacobian of $h_m(w^0)$ can be calculated as follows:

$$\mathrm{J}_{h_m(w^0)} = \sum_{s=1}^{m-1} \sum_{i=1}^{\bar{N}(s)} \phi_{j^*}^{(i)}(s)\, \phi_{j^*}^{'\,(i)}(s)\, \dot{\sigma}\big(\langle\, \phi_{j^*}^{(i)}(s), w^0 \rangle\big) + \kappa I.$$

Let $G_m(w^0) = \int_0^1 \mathrm{J}_{h_m(w^0)}\ d\epsilon$ and recall that $c_\sigma = \inf_{w,\phi_{kj}} \dot{\sigma}\big(\langle\, \phi_{kj}, w\rangle\big)$. Then, we have $G_m(w^0) \succeq c_\sigma\, U_m \succeq c_\sigma(\gamma I) \succ 0$, where U_m is the design matrix and $\kappa = c_\sigma \gamma > 0$. This implies that the matrix $G_m(w^0)$ is positive definite and non-singular. Then, we have:

$$w - \hat{w}(m) = G_m^{-1}(w^0)\big(h_m(w) - h_m(\hat{w}(m))\big).$$

The proof is completed by the following:

$$\begin{aligned}
\big|\sigma\big(\langle\, \phi_{j^*}^{(i)}(m), w\rangle\big) - \sigma\big(\langle\, \phi_{j^*}^{(i)}(m), \hat{w}(m)\rangle\big)\big| &\leq \frac{1}{4}\left|\Big\langle \phi_{j^*}^{(i)}(m), w - \hat{w}(m) \Big\rangle\right| \\
&= \frac{1}{4}\left|\Big\langle \phi_{j^*}^{(i)}(m), G_m^{-1}(w^0)\big(h_m(w) - h_m(\hat{w}(m))\big) \Big\rangle\right| \\
&\leq \frac{1}{4}\left\|\phi_{j^*}^{(i)}(m)\right\|_{G_m^{-1}(w^0)} \|h_m(w) - h_m(\hat{w}(m))\|_{G_m^{-1}(w^0)} \\
&\leq \frac{1}{4c_\sigma}\left\|\phi_{j^*}^{(i)}(m)\right\|_{U_m^{-1}} \|h_m(w) - h_m(\hat{w}(m))\|_{U_m^{-1}},
\end{aligned}$$

where the first inequality holds by the *Lipschitz* property of the logistic function for which the Lipschitz constant is 1/4. The equality holds by the mean value theorem. The last inequality holds by $G_m^{-1}(w^0) \preceq c_\sigma^{-1}\ U_m^{-1}$, which implies that $\|x\|_{G_m^{-1}(w^0)} \leq \frac{1}{\sqrt{c_\sigma}}\ \|x\|_{U_m^{-1}}$ for any vector $x \in \mathbb{R}^d$. □

Lemma III.6 (**Bound on Summation of Feature Vectors**). *Let* $\{\phi_{j^*}^{(i)}(m)\}_{m\in\mathcal{M}, i\in\bar{N}(m)}$ *be a sequence of feature vectors in* $\mathbb{R}^d$*. Then, the following holds:*

$$\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)} \left\|\phi_{j^*}^{(i)}(m)\right\|_{U_m^{-1}} \leq \sqrt{10}\left(\sqrt{dM\bar{N}_{\max}} + \bar{N}_{\max}\frac{d}{\sqrt{\gamma}}\right)\log\left(\frac{\gamma + T}{\gamma}\right).$$

Proof. We adapt the proof proposed by [58] for our regularized setting with varying batch size.

Using the Cauchy-Schwarz inequality, we have:

$$\sum_{m=1}^{M}\sum_{i=1}^{\bar{N}(m)}\left\|\phi_{j^*}^{(i)}(m)\right\|_{U_m^{-1}} \leq \sqrt{\bar{N}_{\max}}\sum_{m=1}^{M}\sqrt{\sum_{i=1}^{\bar{N}(m)}\left\|\phi_{j^*}^{(i)}(m)\right\|_{U_m^{-1}}^2} = \sqrt{\bar{N}_{\max}}\sum_{m=1}^{M}\sqrt{\operatorname{tr}(U_m^{-1}E_m)}, \tag{3.29}$$

where $E_m = \sum_{i=1}^{\bar{N}(m)} \phi_{j^*}^{(i)}(m)\, \phi_{j^*}^{\prime(i)}(m)$.

Similar to Step 2 of Lemma III.3, we have:

$$\begin{aligned}
&\sum_{m=1}^{M}\sqrt{\operatorname{tr}\left(U_m^{-1}E_m\right)} \\
&\leq \sqrt{10}\sum_{m=1}^{M}\sqrt{\sum_{j=1}^{d}\frac{\nu_{m+1,j}-\nu_{m,j}}{\nu_{m+1,j}}} + \sqrt{10}\left(\sum_{m=1}^{M}\sqrt{\sum_{j=1}^{d}\frac{\nu_{m+1,j}-\nu_{m,j}}{\nu_{m,j}}} - \sum_{m=1}^{M}\sqrt{\sum_{j=1}^{d}\frac{\nu_{m+1,j}-\nu_{m,j}}{\nu_{m+1,j}}}\right),
\end{aligned} \tag{3.30}$$

where $\nu_{m,j}$ is the j^{th} eigenvalue of U_m.

First, we upper bound the first term in (3.30). Using the Cauchy-Schwarz inequality, we have:

$$\sum_{m=1}^{M}\sqrt{\sum_{j=1}^{d}\frac{\nu_{m+1,j}-\nu_{m,j}}{\nu_{m+1,j}}} \leq \sqrt{M\sum_{m=1}^{M}\sum_{j=1}^{d}\frac{\nu_{m+1,j}-\nu_{m,j}}{\nu_{m+1,j}}} \leq \sqrt{dM\log\left(\frac{\gamma+T}{\gamma}\right)}, \tag{3.31}$$

where the last inequality holds by Step 2 of Lemma III.3.

Next, we upper bound the second term in (3.30) as follows:

$$\begin{aligned}
&\sum_{m=1}^{M}\sqrt{\sum_{j=1}^{d}\frac{\nu_{m+1,j}-\nu_{m,j}}{\nu_{m,j}}} - \sum_{m=1}^{M}\sqrt{\sum_{j=1}^{d}\frac{\nu_{m+1,j}-\nu_{m,j}}{\nu_{m+1,j}}} \leq \sum_{m=1}^{M}\frac{\sum_{j=1}^{d}\frac{(\nu_{m+1,j}-\nu_{m,j})^2}{\nu_{m,j}\,\nu_{m+1,j}}}{\sqrt{\sum_{j=1}^{d}\frac{\nu_{m+1,j}-\nu_{m,j}}{\nu_{m,j}}}} \\
&= \sum_{m=1}^{M}\frac{\sum_{j=1}^{d}\frac{(\nu_{m+1,j}-\nu_{m,j})^{1/2}}{\nu_{m,j}^{1/2}}\,\frac{(\nu_{m+1,j}-\nu_{m,j})^{3/2}}{\nu_{m,j}^{1/2}\,\nu_{m+1,j}}}{\sqrt{\sum_{j=1}^{d}\frac{\nu_{m+1,j}-\nu_{m,j}}{\nu_{m,j}}}} \leq \sum_{m=1}^{M}\frac{\sqrt{\sum_{j=1}^{d}\frac{\nu_{m+1,j}-\nu_{m,j}}{\nu_{m,j}}}\sqrt{\sum_{j=1}^{d}\frac{(\nu_{m+1,j}-\nu_{m,j})^3}{\nu_{m,j}\,\nu_{m+1,j}^2}}}{\sqrt{\sum_{j=1}^{d}\frac{\nu_{m+1,j}-\nu_{m,j}}{\nu_{m,j}}}} \\
&= \sum_{m=1}^{M}\sqrt{\sum_{j=1}^{d}\frac{\left(\nu_{m+1,j}-\nu_{m,j}\right)^3}{\nu_{m,j}\,\nu_{m+1,j}^2}} \leq \sqrt{\frac{\bar{N}_{\max}}{\gamma}\sum_{m=1}^{M}\sum_{j=1}^{d}\frac{\nu_{m+1,j}-\nu_{m,j}}{\nu_{m+1,j}}} \\
&\leq \sqrt{\frac{\bar{N}_{\max}}{\gamma}\,d\log\left(\frac{\gamma+T}{\gamma}\right)},
\end{aligned} \tag{3.32}$$

where the first inequality holds by $\sqrt{a}-\sqrt{b} \leq \frac{a-b}{\sqrt{a}}$ for $a, b \in \mathbb{R}$. The second inequality holds by the Cauchy-Schwarz inequality. The third inequality holds by having $\nu_{m,j} \geq \nu_{1,j} = \gamma$ and $\nu_{m+1,j} - \nu_{m,j} \leq \bar{N}_{\max}$.

Replacing the results obtained by (3.31) and (3.32) into (3.30) yields the following:

$$\begin{aligned}\sum_{m=1}^{M} \sqrt{\operatorname{tr}\left(U_m^{-1} E_m\right)} &\leq \sqrt{10}\left(\sqrt{dM \log\left(\frac{\gamma+T}{\gamma}\right)} + \sqrt{\frac{\bar{N}_{\max}}{\gamma} d \log\left(\frac{\gamma+T}{\gamma}\right)}\right)\\ &\leq \sqrt{10}\left(\sqrt{dM} + \sqrt{\frac{\bar{N}_{\max}}{\gamma} d}\right) \log\left(\frac{\gamma+T}{\gamma}\right).\end{aligned}$$

Finally, plugging the above result into (3.29), completes the proof. □

The following Lemma is stated without the proof. We refer interested readers to Theorem 2.1 in [61] for details.

Lemma III.7 (**A Tail Bound for Poisson Random Variables**). *Let X be a Poisson random variable with mean $\theta > 0$. Then, for any $x > 0$, we have:*

$$\mathbb{P}\left(X \geq \theta + x\right) \leq e^{-\frac{x^2}{2\theta} g\left(\frac{x}{\theta}\right)},$$

where $g : [-1, \infty] \to \mathbb{R}$ is a function defined by $g(u) = 2\,\frac{(1+u)\log(1+u)-u}{u^2}$.
As $g(\frac{x}{\theta}) \geq \frac{1}{1+\frac{x}{\theta}}$ for every $x > 0$, we have:

$$\mathbb{P}\left(X \geq \theta + x\right) \leq e^{-\frac{x^2}{2(\theta+x)}}.$$

Remark. *Let T be a Poisson random variable with mean $\bar{T}$. Then, it is upper bounded by $\bar{T} - \log(\delta) + \sqrt{\left(\log(\delta)\right)^2 - 2\bar{T}\log(\delta)}$ with probability at least $1-\delta$.*

CHAPTER IV

Conclusions and Future Research

4.1 Summary and Conclusions

This dissertation introduced a new approach to solve problems for which there is a need of joint contextual learning and resource allocation under different types of limited resources. We developed data-driven and personalized decision-making frameworks to address practical OR/OM problems. We focused on cutting edge problems in healthcare delivery and hospital operations, each of which has its own nuance structure that spurred several innovations and tailored methods.

Chapter II focused on addressing the problem of online resource allocation with learning under single-use resources. In this problem, a decision-maker needs to assign arbitrary sequentially arriving users to a resource, where the reward of each assignment and resource consumption are unknown, and feedback is received with delay. This problem structure can be frequently observed in a broad range of OR/OM problems. We introduced a new framework that judiciously synergizes online learning theory with a broad class of online resource allocation mechanisms when the system has single-use resources. We designed online algorithms with a performance guarantee in terms of the notion of regret that bridges the gap between online learning and resource allocation. Our analysis demonstrates the possibility of bounding the performance of this class of online algorithms by a seamless integration of competitive ratio bounds for online resource allocation algorithms and Bayesian regret bounds for contextual learning algorithms. We showed that a bridging technique allows decomposing the regret into two types of loss. The first type stems from learning with delayed feedback, which scales sub-linearly in the number of arrivals over the planning horizon. A useful facet of our analysis is that the cost of learning under delayed feedback is an additive term in the contextual learning loss, which goes to zero as the delay time goes to zero. The second type stems from resource allocation under no information on future arrivals, where it is bounded by a fixed fraction of the expected reward obtained by the clairvoyant benchmark.

We also investigated the advance scheduling problem, which can be viewed as an application of our generic framework. A major difference between our general resource allocation and advance scheduling problems is that the latter provides multi-day scheduling and captures the no-show behavior of customers. In this problem, customers may not show up for the scheduled service after being assigned to a server and a service date. This adds on an additional layer of complexity in learning the match quality because the match quality feedback of a customer cannot be observed at all if the customer does not show up on the service date. We addressed this additional complexity by constructing a new confidence bound with an extra additive logarithmic term in the number of arrivals. This additive term can be viewed as the cost of learning in this setting. We used the advance scheduling algorithm to provide an appointment scheduling platform taking into account the patient-provider match quality, visit time, possibility of no-show, and the availability of providers. The empirical performance of our methodology was assessed using the data set from our partner hospital. The results demonstrate that our algorithm achieves near-optimal performance. We find that the greedy and the commonly used first-come-first-served policies have poor performance when there are scarce resources. In our case study, the difference in performance is up to 30%. The results also demonstrate that our advance scheduling algorithm can handle no-shows and delayed feedback very well. As a key insight, we highlight the importance of capturing the delayed feedback both in modeling and theoretical analysis as it clearly impacts both the empirical and theoretical results.

Motivated by our successful methodology for the joint learning and resource allocation under single-use resources, in Chapter III, we extended it to handle reusable resources. In particular, we studied the problem of optimizing care unit placement decisions in hospitals. We designed an online learning algorithm that addressed the high variability in patients' health and high utilization of hospital beds. We proposed a mechanism for batch learning under delayed feedback. We also designed and included a policy guide model which strikes a trade-off between the (i) benefits of assigning acute patients to high-level beds and (ii) sub-optimal use of such limited resources for patients who may not benefit sufficiently to warrant having them. Using the assumption of known arrival rates, our modeling induces a loss network system. A key part of our theoretical analysis is showing that this loss can be bounded as a fraction of the total expected reward obtained by our algorithm. Another part of our theoretical analysis demonstrates that the batch learning loss with delay is again sub-linear and vanishes as the delay goes to zero.

Our results indicate that implementing our algorithm with the sole aim of reducing readmissions has the potential to improve the hospital's admission policy via decreasing the readmission rate (up to 10%), managing congestion in different care units, and judiciously prioritizing the critical beds. We note that the current hospital's policy is designed with the aim of reducing the mortality risk. Our focus on hospital readmission was a way to identify the limits to which an online algorithm for care unit placement can improve the readmission rate. Future research is needed to appropriately blend the readmission reduction objective with other objectives and constraints that the hospital may have. A key insight of our case study is that the revealed information on prior patient responses can be used to improve care unit placement decisions by reducing the patient exposure to less effective decisions and exploring promising care unit placements. This is partly due to the distinctive feature of our approach (on the fly strategy), which is effective to learn fast when the feedback of prior care unit assignments is delayed in time. Our results suggest several underlying reasons for the high success rate of decisions provided by our algorithm compared to the hospital's policy. Apart from estimating the risk of readmission adaptively, one reason is very likely that the current policy of the hospital is not able to properly account for the opportunity cost of using each bed type. Our approach accounts for the opportunity cost of using an available bed or saving it for complex patient arrivals in the future. This is done by providing a time interval- and type-dependent admission policy by leveraging the estimated congestion in the network of units and the future patient arrival pattern.

4.2 Future Research

In summary, this dissertation addressed the fundamental question of how to design online decision-making algorithms that operate under limited resources; however, several avenues of research remain that can expand on this dissertation. We believe these future research questions can significantly contribute to both methodological and practical aspects of this area.

With the advancement of artificial intelligence, decision-making for numerous aspects of our daily lives is being outsourced to artificial intelligence algorithms. These algorithms can make a significant number of decisions over a short period, where a small disparity in most applications may lead to discrimination that can have a huge impact on society. Policy-makers and regulators have recently expressed fears about the potentially discriminatory impact of these algorithms. This calls for fairness considerations to avoid the danger of inadvertently encoding bias into automated decisions.

It is well-known that exploration is a key ingredient of online decision-making algorithms. Of course, exploration is associated with a cost relevant to making decisions that may eventually reveal to be sub-optimal. It has been shown that the commonly used mechanisms to encourage exploration, such as posterior sampling and upper confidence bound methods, are effective in reducing the overall regret. However, these methods lack a systematic procedure to ensure their exploration will be fair for users/patients. In Chapters II and III, incorporating fairness into our learning algorithm will be a promising future research avenue. To do so, we need to clearly define the notion of fairness. Although several notions of fairness and explainability have been introduced in the machine learning literature, we believe solving healthcare problems necessitates careful thought when defining a fairness measure. For example, one can define a fairness notion to ensure that the cost of exploration is evenly distributed to different groups of patients. An interesting avenue for future research could be around identifying an appropriate fairness notion that is also explainable, and designing algorithms capable of including such fairness considerations. Designing fair algorithms to effectively capture the trade-off between fairness and achieving overall low regret is another promising direction for future research.

In Chapter II, we studied a problem that requires online sequential decision-making under single-use resources. We designed two algorithms to make online resource allocation without any information on future arrivals. Chapter III focused on a problem that requires online sequential decision-making under reusable resources. Assuming that the number of arrivals follows Poisson distributions with known arrival rates, we incorporated a policy guide into our algorithm to allocate limited resources to hedge against future arrivals who potentially can benefit more from those resources. The assumption of the stochastic arrival process is shown to be a valid assumption for many applications under normal situations. However, there are some other situations in which the historical data on future arrivals can become obsolete because of the rapid changes in a system. That is, the sequence of future arrivals becomes arbitrary and even possibly chosen adversarially. For example, the historical hospital demand data might become obsolete when an unexpected crisis hits, and the health care system must adjust accordingly. Thus, the following natural question arises: how to design an algorithm to sequentially allocate limited reusable resources when there is a lack of information on future demand? Relaxing the assumption of known arrival rates could be an interesting frontier for future research to explore.

BIBLIOGRAPHY

[1] Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, pages 2312–2320, 2011.

[2] Alekh Agarwal, Daniel Hsu, Satyen Kale, John Langford, Lihong Li, and Robert Schapire. Taming the monster: A fast and simple algorithm for contextual bandits. In *International Conference on Machine Learning*, pages 1638–1646, 2014.

[3] Shipra Agrawal and Nikhil Devanur. Linear contextual bandits with knapsacks. In *Advances in Neural Information Processing Systems*, pages 3450–3458, 2016.

[4] Shipra Agrawal and Nikhil R Devanur. Bandits with concave rewards and convex knapsacks. In *Proceedings of the fifteenth ACM conference on Economics and computation*, pages 989–1006, 2014.

[5] Shipra Agrawal, Nikhil R Devanur, and Lihong Li. An efficient algorithm for contextual bandits with knapsacks, and an extension to concave objectives. In *Conference on Learning Theory*, pages 4–18, 2016.

[6] Shipra Agrawal and Navin Goyal. Thompson sampling for contextual bandits with linear payoffs. In *International Conference on Machine Learning*, pages 127–135, 2013.

[7] 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.

[8] Peter Auer. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3(Nov):397–422, 2002.

[9] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2):235–256, 2002.

[10] A. Badanidiyuru, R. Kleinberg, and A. Slivkins. Bandits with knapsacks. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 207–216, 2013.

[11] Ashwinkumar Badanidiyuru, Robert Kleinberg, and Aleksandrs Slivkins. Bandits with knapsacks. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 207–216. IEEE, 2013.

[12] Ashwinkumar Badanidiyuru, John Langford, and Aleksandrs Slivkins. Resourceful contextual bandits. In *Conference on Learning Theory*, pages 1109–1134, 2014.

[13] 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.

[14] Hamsa Bastani and Mohsen Bayati. Online decision making with high-dimensional covariates. *Operations Research*, 68(1):276–294, 2020.

[15] Hamsa Bastani, Mohsen Bayati, and Khashayar Khosravi. Mostly exploration-free algorithms for contextual bandits. *arXiv preprint arXiv:1704.09011*, 2017.

[16] Nicole Bäuerle. Asymptotic optimality of tracking policies in stochastic networks. *The Annals of Applied Probability*, 10(4):1065–1083, 2000.

[17] Jochanan Benbassat and Mark Taragin. Hospital readmissions as a measure of quality of health care: advantages and limitations. *Archives of internal medicine*, 160(8):1074–1081, 2000.

[18] Dimitris Bertsimas and Nathan Kallus. From predictive to prescriptive analytics. *Management Science*, 66(3):1025–1044, 2020.

[19] Dimitris Bertsimas, Agni Orfanoudaki, and Rory B Weiner. Personalized treatment for coronary artery disease patients: A machine learning approach. *arXiv preprint arXiv:1910.08483*, 2019.

[20] Dimitris Bertsimas, Agni Orfanoudaki, and Rory B Weiner. Personalized treatment for coronary artery disease patients: a machine learning approach. *Health Care Management Science*, 23(4):482–506, 2020.

[21] Kostas Bimpikis and Mihalis G Markakis. Learning and hierarchies in service systems. *Management Science*, 65(3):1268–1285, 2019.

[22] Ilai Bistritz, Zhengyuan Zhou, Xi Chen, Nicholas Bambos, and Jose Blanchet. Online exp3 learning in adversarial bandits with delayed feedback. In *Advances in Neural Information Processing Systems*, pages 11349–11358, 2019.

[23] David B Brown and Jingwei Zhang. Dynamic programs with shared resources and signals: Dynamic fluid policies and asymptotic optimality. *Operations Research*, 2021.

[24] Lawrence Brown, Noah Gans, Avishai Mandelbaum, Anat Sakov, Haipeng Shen, Sergey Zeltyn, and Linda Zhao. Statistical analysis of a telephone call center: A queueing-science perspective. *Journal of the American statistical association*, 100(469):36–50, 2005.

[25] Niv Buchbinder, Kamal Jain, and Joseph Naor. Online primal-dual algorithms for maximizing ad-auctions revenue. *Algorithms–ESA 2007*, pages 253–264, 2007.

[26] Cagatay Catal and Mehmet Nangir. A sentiment classification model based on multiple classifiers. *Applied Soft Computing*, 50:135–141, 2017.

[27] Carri W Chan, Vivek F Farias, Nicholas Bambos, and Gabriel J Escobar. Optimizing intensive care unit discharge decisions with patient readmissions. *Operations research*, 60(6):1323–1341, 2012.

[28] Carri W Chan, Linda V Green, Suparerk Lekwijit, Lijian Lu, and Gabriel Escobar. Assessing the impact of service level when customer needs are uncertain: An empirical investigation of hospital step-down units. *Management Science*, 65(2):751–775, 2018.

[29] Olivier Chapelle and Lihong Li. An empirical evaluation of thompson sampling. In *Advances in neural information processing systems*, pages 2249–2257, 2011.

[30] Jinsheng Chen, Jing Dong, and Pengyi Shi. Optimal routing under demand surges: The value of future arrival rates. *Available at SSRN 3980227*, 2021.

[31] Lena M Chen, Edward H Kennedy, Anne Sales, and Timothy P Hofer. Use of health it for higher-value critical care. *New England Journal of Medicine*, 368(7):594–597, 2013.

[32] Lena M Chen, Marta Render, Anne Sales, Edward H Kennedy, Wyndy Wiitala, and Timothy P Hofer. Intensive care unit admitting patterns in the veterans affairs health care system. *Archives of internal medicine*, 172(16):1220–1226, 2012.

[33] Xi Chen, Zachary Owen, Clark Pixton, and David Simchi-Levi. A statistical learning approach to personalization in revenue management. *Management Science*, 2021.

[34] Yiwei Chen, Retsef Levi, and Cong Shi. Revenue management of reusable resources with advanced reservations. *Production and Operations Management*, 26(5):836–859, 2017.

[35] Wang Chi Cheung, Will Ma, David Simchi-Levi, and Xinshang Wang. Inventory balancing with online learning. *arXiv preprint arXiv:1810.05640; forthcoming in Management Science*, 2022.

[36] Wang Chi Cheung, David Simchi-Levi, and Ruihao Zhu. Hedging the drift: Learning to optimize under non-stationarity. *arXiv preprint arXiv:1903.01461*, 2019.

[37] 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*, pages 208–214, 2011.

[38] Mark E Cowen, Jennifer L Czerwinski, Patricia J Posa, Elizabeth Van Hoek, James Mattimore, Lakshmi K Halasyamani, and Robert L Strawderman. Implementation of a mortality prediction rule for real-time decision making: Feasibility and validity. *Journal of hospital medicine*, 9(11):720–726, 2014.

[39] Mark E Cowen, Robert L Strawderman, Jennifer L Czerwinski, Mary Jo Smith, and Lakshmi K Halasyamani. Mortality predictions on admission as a context for organizing care activities. *Journal of hospital medicine*, 8(5):229–235, 2013.

[40] JG Dai and Pengyi Shi. Inpatient overflow: An approximate dynamic programming approach. *Manufacturing & Service Operations Management*, 21(4):894–911, 2019.

[41] Varsha Dani, Thomas P Hayes, and Sham M Kakade. Stochastic linear optimization under bandit feedback. *Annual Conference on Learning Theory*, pages 355–366, 2008.

[42] Arlean Dean, Mohammad Zhalechian, and Mark P Van Oyen. Dynamic care unit placements under unknown demand with learning. *Available at SSRN*, 2022.

[43] 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.

[44] Jing Dong, Pengyi Shi, Fanyin Zheng, and Xin Jin. Off-service placement in inpatient ward network: Resource pooling versus service slowdown. *Columbia Business School Research Paper Forthcoming*, 2019.

[45] Miroslav Dudik, Daniel Hsu, Satyen Kale, Nikos Karampatziakis, John Langford, Lev Reyzin, and Tong Zhang. Efficient optimal learning for contextual bandits. *arXiv preprint arXiv:1106.2369*, 2011.

[46] Adam N Elmachtoub, Ryan McNellis, Sechan Oh, and Marek Petrik. A practical method for solving contextual bandit problems using decision trees. *arXiv preprint arXiv:1706.04687*, 2017.

[47] 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.

[48] Yiding Feng, Rad Niazadeh, and Amin Saberi. Near-optimal bayesian online assortment of reusable resources. Working Paper, Chicago Booth Business School, Chicago, IL, 2020.

[49] Kris Johnson Ferreira, David Simchi-Levi, and He Wang. Online network revenue management using thompson sampling. *Operations research*, 66(6):1586–1602, 2018.

[50] Sarah Filippi, Olivier Cappe, Aurélien Garivier, and Csaba Szepesvári. Parametric bandits: The generalized linear case. In *Advances in Neural Information Processing Systems*, pages 586–594, 2010.

[51] Guillermo Gallego, Garud Iyengar, Robert Phillips, and Abhay Dubey. Managing flexible products on a network. *Available at SSRN 3567371*, 2004.

[52] Geoffrey Gerhardt, A Yemane, P Hickman, A Oelschlaeger, E Rollins, and N Brennan. Data shows reduction in medicare hospital readmission rates during 2012. *Medicare Medicaid Res Rev*, 3(2):E1–E11, 2013.

[53] Negin Golrezaei, Hamid Nazerzadeh, and Paat Rusmevichientong. Real-time optimization of personalized assortments. *Management Science*, 60(6):1532–1551, 2014.

[54] Xiao-Yue Gong, Vineet Goyal, Garud N Iyengar, David Simchi-Levi, Rajan Udwani, and Shuangyu Wang. Online assortment optimization with reusable resources. *Management Science*, 2021.

[55] Diwakar Gupta and Brian Denton. Appointment scheduling in health care: Challenges and opportunities. *IIE transactions*, 40(9):800–819, 2008.

[56] 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.

[57] Neil A Halpern, Stephen M Pastores, Howard T Thaler, and Robert J Greenstein. Critical care medicine use and cost among medicare beneficiaries 1995–2000: Major discrepancies between two united states federal medicare databases. *Critical care medicine*, 35(3):692–699, 2007.

[58] Yanjun Han, Zhengqing Zhou, Zhengyuan Zhou, Jose Blanchet, Peter W Glynn, and Yinyu Ye. Sequential batch learning in finite-action linear contextual bandits. *arXiv preprint arXiv:2004.06321*, 2020.

[59] Jonathan E Helm, Adel Alaeddini, Jon M Stauffer, Kurt M Bretthauer, and Ted A Skolarus. Reducing hospital readmissions by integrating empirical prediction with resource optimization. *Production and Operations Management*, 25(2):233–257, 2016.

[60] Jonathan E Helm and Mark P Van Oyen. Design and optimization methods for elective hospital admissions. *Operations Research*, 62(6):1265–1282, 2014.

[61] Svante Janson, Andrzej Rucinski, and Tomasz Luczak. *Random graphs*. John Wiley & Sons, 2011.

[62] Ramesh Johari, Vijay Kamble, and Yash Kanoria. Know your customer: Multi-armed bandits with capacity constraints. *arXiv preprint arXiv:1603.04549*, 2016.

[63] Ramesh Johari, Vijay Kamble, and Yash Kanoria. Matching while learning. *Operations Research*, 69(2):655–681, 2021.

[64] Pooria Joulani, Andras Gyorgy, and Csaba Szepesvári. Online learning under delayed feedback. In *International Conference on Machine Learning*, pages 1453–1461, 2013.

[65] Robert S Kaplan and Michael E Porter. How to solve the cost crisis in health care. *Harvard Business Review*, 89(9):46–52, 2011.

[66] Nathaniel Kell and Debmalya Panigrahi. Online budgeted allocation with general budgets. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, pages 419–436, 2016.

[67] Esmaeil Keyvanshokooh, Cong Shi, and Mark P Van Oyen. Online advance scheduling with overtime: A primal-dual approach. *Manufacturing & Service Operations Management*, 23(1):246–266, 2021.

[68] Esmaeil Keyvanshokooh, Mohammad Zhalechian, Cong Shi, Mark P Van Oyen, and Pooyan Kazemian. Contextual learning with online convex optimization with applications to medical decision-making. *Available at SSRN 3501316*, 2019.

[69] Song-Hee Kim, Carri W Chan, Marcelo Olivares, and Gabriel Escobar. Icu admission control: An empirical study of capacity allocation and its implication for patient outcomes. *Management Science*, 61(1):19–38, 2014.

[70] Song-Hee Kim, Ponni Vel, Ward Whitt, and Won Chul Cha. Poisson and non-poisson properties in appointment-generated arrival processes: The case of an endocrinology clinic. *Operations Research Letters*, 43(3):247–253, 2015.

[71] Subhashini Krishnasamy, Rajat Sen, Ramesh Johari, and Sanjay Shakkottai. Regret of queueing bandits. *Advances in Neural Information Processing Systems*, 29, 2016.

[72] Subhashini Krishnasamy, Rajat Sen, Ramesh Johari, and Sanjay Shakkottai. Learning unknown service rates in queues: A multiarmed bandit approach. *Operations Research*, 69(1):315–330, 2021.

[73] Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

[74] Yanzhe Murray Lei and Stefanus Jasin. Real-time dynamic pricing for revenue management with reusable resources, advance reservation, and deterministic service time requirements. *Forthcoming in Operations Research*, 2020.

[75] Retsef Levi and Ana Radovanović. Provably near-optimal lp-based policies for revenue management in systems with reusable resources. *Operations Research*, 58(2):503–507, 2010.

[76] Lihong Li, Wei Chu, John Langford, and Robert E Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 661–670. ACM, 2010.

[77] 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*, pages 2071–2080. JMLR. org, 2017.

[78] 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.

[79] Qian Liu and Garrett Van Ryzin. On the choice-based linear programming model for network revenue management. *Manufacturing & Service Operations Management*, 10(2):288–310, 2008.

[80] Constantinos Maglaras. Discrete-review policies for scheduling stochastic networks: Trajectory tracking and fluid-scale asymptotic optimality. *The Annals of Applied Probability*, 10(3):897–929, 2000.

[81] 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.

[82] Amirhossein Meisami, Jivan Deglise-Hawkinson, Mark E Cowen, and Mark P Van Oyen. Data-driven optimization methodology for admission control in critical care units. *Health care management science*, 22(2):318–335, 2019.

[83] Timothy E Miller, Julie K Thacker, William D White, Christopher Mantyh, John Migaly, Juying Jin, Anthony M Roche, Eric L Eisenstein, Rex Edwards, Kevin J Anstrom, et al. Reduced length of hospital stay in colorectal surgery after implementation of an enhanced recovery protocol. *Anesthesia & Analgesia*, 118(5):1052–1061, 2014.

[84] Yonatan Mintz, Anil Aswani, Philip Kaminsky, Elena Flowers, and Yoshimi Fukuoka. Non-stationary bandits with habituation and recovery dynamics. *arXiv preprint arXiv:1707.08423*, 2017.

[85] David P Morton and R Kevin Wood. On a stochastic knapsack problem and generalizations. In *Advances in computational and stochastic optimization, logic programming, and heuristic search*, pages 149–168. Springer, 1998.

[86] 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.

[87] Annette M O'Connor, Hilary A Llewellyn-Thomas, and Ann Barry Flood. Modifying unwarranted variations in health care: Shared decision making using patient decision aids: A review of the evidence base for shared decision making. *Health Affairs*, 23(Suppl2):VAR–63, 2004.

[88] Zachary Owen and David Simchi-Levi. Price and assortment optimization for reusable resources. *Available at SSRN 3070625*, 2018.

[89] Xin Pan, Jie Song, Jingtong Zhao, and Van-Anh Truong. Online contextual learning with perishable resources allocation. *IISE Transactions*, pages 1–15, 2020.

[90] Jonathan Patrick, Martin L Puterman, and Maurice Queyranne. Dynamic multipriority patient scheduling for a diagnostic resource. *Operations research*, 56(6):1507–1525, 2008.

[91] Ciara Pike-Burke, Shipra Agrawal, Csaba Szepesvari, and Steffen Grunewalder. Bandits with delayed, aggregated anonymous feedback. *arXiv preprint arXiv:1709.06853*, 2017.

[92] Meghan Prin and Hannah Wunsch. The role of stepdown beds in hospital care. *American journal of respiratory and critical care medicine*, 190(11):1210–1216, 2014.

[93] D Reis Miranda and M Jegers. Monitoring costs in the icu: a search for a pertinent methodology. *Acta Anaesthesiologica Scandinavica*, 56(9):1104–1113, 2012.

[94] Paat Rusmevichientong and John N Tsitsiklis. Linearly parameterized bandits. *Mathematics of Operations Research*, 35(2):395–411, 2010.

[95] Daniel Russo and Benjamin Van Roy. Learning to optimize via posterior sampling. *Mathematics of Operations Research*, 39(4):1221–1243, 2014.

[96] Saied Samiedaluie, Beste Kucukyazici, Vedat Verter, and Dan Zhang. Managing patient admissions in a neurology ward. *Operations Research*, 65(3):635–656, 2017.

[97] Virag Shah, Lennart Gulikers, Laurent Massoulié, and Milan Vojnović. Adaptive matching for expert systems with uncertain task types. *Operations Research*, 68(5):1403–1424, 2020.

[98] Amir Shmueli, Charles L Sprung, and Edward H Kaplan. Optimizing admissions to an intensive care unit. *Health Care Management Science*, 6(3):131–136, 2003.

[99] Aleksandrs Slivkins. Introduction to multi-armed bandits. *arXiv preprint arXiv:1904.07272*, 2019.

[100] Clifford Stein, Van-Anh Truong, and Xinshang Wang. Advance service reservations with heterogeneous customers. *Management Science*, 2019.

[101] Clifford Stein, Van-Anh Truong, and Xinshang Wang. Advance service reservations with heterogeneous customers. *Management Science*, 66(7):2929–2950, 2020.

[102] Robert H Thiele, Kathleen M Rea, Florence E Turrentine, Charles M Friel, Taryn E Hassinger, Bernadette J Goudreau, Bindu A Umapathi, Irving L Kron, Robert G Sawyer, Traci L Hedrick, et al. Standardization of care: impact of an enhanced recovery protocol on length of stay, complications, and direct costs after colorectal surgery. *Journal of the American College of Surgeons*, 220(4):430–443, 2015.

[103] Van-Anh Truong. Optimal advance scheduling. *Management Science*, 61(7):1584–1597, 2015.

[104] Richard Van Slyke and Yi Young. Finite horizon stochastic knapsacks with applications to yield management. *Operations Research*, 48(1):155–172, 2000.

[105] Claire Vernade, Olivier Cappé, and Vianney Perchet. Stochastic bandit models for delayed conversions. *arXiv preprint arXiv:1706.09186*, 2017.

[106] Wen-Ya Wang and Diwakar Gupta. Adaptive appointment systems with patient preferences. *Manufacturing & Service Operations Management*, 13(3):373–389, 2011.

[107] Xinshang Wang, Van-Anh Truong, and David Bank. Online advance admission scheduling for services with customer preferences. *arXiv preprint arXiv:1805.10412*, 2018.

[108] Yingfei Wang, Chu Wang, and Warren Powell. The knowledge gradient for sequential decision making with stochastic binary feedbacks. In *International Conference on Machine Learning*, pages 1138–1147, 2016.

[109] Gabriel Zayas-Caban, Stefanus Jasin, and Guihua Wang. An asymptotically optimal heuristic for general nonstationary finite-horizon restless multi-armed, multi-action bandits. *Advances in Applied Probability*, 51(3):745–772, 2019.

[110] Mohammad Zhalechian, Esmaeil Keyvanshokooh, Cong Shi, and Mark P Van Oyen. Personalized hospital admission control: a contextual learning approach. *Available at SSRN 3653433*, 2020.

[111] Mohammad Zhalechian, Esmaeil Keyvanshokooh, Cong Shi, and Mark P Van Oyen. Online resource allocation with personalized learning. *Operations Research*, 2022.

[112] Zhengyuan Zhou, Renyuan Xu, and Jose Blanchet. Learning in generalized linear contextual bandits with stochastic delays. In *Advances in Neural Information Processing Systems*, pages 5197–5208, 2019.