Algorithms for Online Marketplaces: New Approaches to Order Fulfillment and Recommendation Systems

by

Ayoub Amil

Business Administration
Duke University

Date: March 18, 2024

Approved:

Ali Makhdoumi, Co-Supervisor

Yehua Wei, Co-Supervisor

Aleksandar Pekeč

Kevin H. Shang

Dissertation submitted in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Business Administration
in the Graduate School of Duke University
2024

Abstract

Algorithms for Online Marketplaces: New Approaches to Order Fulfillment and Recommendation Systems

by

Ayoub Amil

Business Administration
Duke University

Date: March 18, 2024

Approved:

Ali Makhdoumi, Co-Supervisor

Yehua Wei, Co-Supervisor

Aleksandar Pekeč

Kevin H. Shang

An abstract of a dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Business Administration in the Graduate School of Duke University
2024

Abstract

This dissertation explores the development and analysis of new algorithms for sequential decision-making under uncertainty, with a focus on optimizing operations and resource allocations within online marketplaces such as e-commerce and rental platforms. The research initially revisits and expands upon the multi-item order fulfillment model, introducing dynamic policies that combine randomized fulfillment strategies, prophet inequalities, and subgradient methods. Our approaches not only achieve asymptotic optimality and strong approximation guarantees in the multi-item fulfillment setting, but also provide insights on how to construct robust policies in scenarios where you have limited resources. The findings in this dissertation introduce a novel approach to the management of resources in complex environments, presenting a nearly optimal framework for developing policies tailored to the complexities of multi-item order fulfillment. Moreover, our analysis can be extended into the domain of rental operations, showcasing the flexibility and broad applicability of our proposed solutions.

In addition, the dissertation addresses the complexities of online recommendation systems through a contextual bandit framework, examining both full-feedback and bandit-feedback settings. By formulating the problem to accommodate arbitrary mappings from user contexts to product feature values, this research provides new online algorithms that effectively minimize regret. The analysis extends to general policy classes, revealing an inherent trade-off between approximation accuracy and

statistical error for a given policy class.

Collectively, this work advances theoretical knowledge in sequential decision-making and algorithm design, providing actionable strategies for improving decision-making processes such as fulfillment and recommendations in online marketplaces.

Dedication

Dedicated to my family and Elizabeth, for your unwavering support and love. This achievement is not just mine, but ours.

Contents

List of Tables

Acknowledgements

Embarking on this doctoral journey has been an enriching experience, filled with intellectual growth, challenges, and memories. It is a path I could not have navigated without the support, guidance, and encouragement of many individuals who have been instrumental in my academic and personal development. I am deeply grateful for their contributions to my journey.

Foremost, I extend my profound gratitude to my advisors, Ali Makhdoumi and Yehua Wei. Ali, who began guiding me during my second year, demonstrated remarkable patience as we derived proofs from scratch, a testament to his dedication and approach to mentorship. His practicality and ability to communicate complex ideas in a straightforward manner have profoundly impacted my approach to research. Ali's skill in connecting disparate topics to reveal a broader understanding of our field has been invaluable. His openness, kindness, and unwavering support have been crucial as I navigated through various academic avenues, allowing me to approach many areas of interest with confidence and a deep sense of curiosity.

Yehua joined as my advisor in my third year, adding a profound depth to my research through his exceptional mathematical intuition and sharp intellect. The afternoons spent with both Ali and Yehua in front of the whiteboard, working through the nuances of the magician problem, were not only academically enriching but also fostered a sense of camaraderie and mutual support. Yehua's dedication to my academic success, combined with his insightful approach and patience, has provided a

nurturing environment that has significantly contributed to my growth as a researcher and as an individual.

Together, Ali and Yehua have been more than advisors; they have been mentors in the truest sense, guiding me not only in my academic pursuits but also in making key life decisions. Their combined wisdom, patience, and encouragement have formed the cornerstone of my success. I will always value our collaborative efforts, especially those afternoons dedicated to challenging problems, as they have been instrumental in my development. Their mentorship guided me through the challenges of research and academic life, and I am forever grateful for their kindness, support, and the profound impact they have had on my journey.

My initial year at Duke was marked by the expert guidance of Alessandro Arlotto and Jiaming Xu. Alessandro, alongside Jiaming, played a pivotal role in my admission to Duke, both advocating for the program's strength in conducting rigorous methodological research. Their support for my exploration of a research problem that bridges their distinct domains was a critical turning point in my academic journey. The emphasis on mathematical rigor that both Alessandro and Jiaming maintained has been an invaluable learning experience, teaching me not just the mechanics of research but the importance of precision and depth in scientific inquiry. I am immensely grateful for their patience, expertise, and the foundational guidance they provided, which has been instrumental in shaping my approach to research.

Additionally, I extend my appreciation to my committee members: Ali Makhdoumi, Yehua Wei, Saša Pekec, and Kevin Shang. My interactions with Saša, right from the outset of my journey at Duke, have been particularly impactful. As the PhD program coordinator, Saša was instrumental in my admission to Duke, effectively highlighting the program's strengths and its alignment with my aspirations for conducting rigorous methodological research. Our interview, which also included Jiaming, profoundly impressed upon me the collaborative and intellectually stimulating environment that

Duke offers. The guidance and insights provided by Saša and Kevin throughout my time on the committee have been invaluable, offering clarity and direction to my research. Their encouragement has significantly bolstered my academic progress and confidence in navigating my research path.

I am also deeply grateful to Ali Tuna and Cagin Uru for the countless hours we spent together working on homework across various classes, including probability, convex optimization, revenue management, and statistical inference on graphs. The collaborative effort and the shared struggles and triumphs have been a significant part of my academic journey, providing both intellectual growth and a sense of community. Special thanks are due to Lin Zhao, whose technical support with simulations on the Duke cluster was invaluable. Beyond his assistance, Lin has been a friend and a top-tier ping-pong partner, along with Jingwei Zhang. These moments of camaraderie were essential to maintaining a balanced and enjoyable life during the rigors of my Ph.D. studies. Furthermore, my appreciation extends to my friend Zexin Cai, who has been a consistent workout partner, especially in the last few months of my Ph.D. His company and the stimulating conversations about future research ideas have been inspiring and invaluable, offering both physical and mental rejuvenation. To my peers, faculty, and everyone else who has been a part of my journey, your support, whether through discussions, feedback, or simply being there, has been a source of strength and motivation. The intellectual environment at Duke, enriched by your contributions, has been instrumental in shaping my academic career. The sense of belonging, the collaborative spirit, and the shared pursuit of knowledge within this community have profoundly impacted my personal and professional development. I am deeply thankful for every discussion, piece of feedback, and moment of support that has been extended to me throughout this journey.

Lastly, my deepest gratitude extends to Elizabeth, my partner, whose support has been nothing short of extraordinary. Words fall short of capturing the full extent

of my appreciation for her. Elizabeth's compassion and unwavering belief in me have been my guiding lights through the highs and lows of this journey. Her presence has been a constant source of comfort, inspiration, and strength. I am profoundly thankful for her love, her understanding, and the countless ways she has enriched my life. Without her support, this journey would have been infinitely more challenging.

To my family, especially my parents and my brothers, Nour and Nasr, your endless support and encouragement have been foundational to my success. My mother's resilience, love, and sacrifices have shaped me in ways that words cannot describe, and my father's guidance, strength, and wisdom have provided a steady hand through all of life's challenges. Their lessons in responsibility, faith, and creativity have been my north star. Nour and Nasr, your brotherhood has been a source of joy, strength, and motivation. Your support and belief in my capabilities have been unwavering, and for that, I am eternally grateful.

This journey, marked by countless milestones, has been made possible by the love, support, and encouragement of Elizabeth, my family, and all who have been a part of my life during these formative years at Duke. I am blessed to have such a remarkable circle of loved ones, whose influence and support have left an indelible mark on my personal and academic growth. Thank you all, from the bottom of my heart, for everything.

1

Introduction

In the rapidly evolving landscape of online retail and digital platforms, the challenges of optimizing real-time decision-making have become paramount for operational efficiency. As e-commerce continues to scale, with projections indicating a rise in sales to over $8 trillion globally by 2027 (Statista Research Department, 2023, 2024), the complexity of managing large amounts of inventories across multiple warehouses and addressing the diverse preferences of consumers in a timely manner have emerged as critical priorities. The multifaceted nature of e-commerce order fulfillment and online recommendations presents a unique set of problems characterized by uncertain demand, finite resources, and unknown consumer preferences. On one hand, the fulfillment of orders in e-commerce involves navigating through a complex and large decision space to minimize expected costs while satisfying inventory constraints. On the other hand, recommendation systems in digital platforms strive to personalize offerings to enhance user experience, grappling with the statistical challenges of predicting consumer behavior. This dissertation investigates these problems through the lens of optimization and probability theory, exploring how dynamic policies and algorithms can be used to achieve near-optimal performance in these uncertain and

data-rich environments.

The primary goal of this dissertation is twofold: firstly, we want to lay down a solid theoretical framework for developing and analyzing algorithms that effectively address some of the challenges faced by online marketplaces, particularly in the areas of e-commerce fulfillment and online recommendations. Secondly, we want to offer insights and practical solutions to the complex decision-making problems within these areas. Specifically, the methodologies and tools we introduce are designed to provide managers with robust decision-making tools that can be confidently applied in practice.

1.1 Multi-Item Order Fulfillment

In Chapters 2 and 3, we revisit and expand upon the multi-item order fulfillment model from the literature. In Chapter 2, we study the case of non-reusable resources (or inventory); while in Chapter 3 we extend our model to cases in which resources can be reused after a random period of time. Chapter 2 specifically addresses the operational challenges associated with non-reusable resources, where the focus is on optimizing inventory management because of limited or finite resources. In this chapter, we reexamine the multi-item order fulfillment framework originally presented by Jasin and Sinha (2015). In particular, we study a dynamic setting in which an e-commerce platform, operating with multiple warehouses (or facilities) and finite inventory, needs to decide how to dispatch orders across their warehouses. The goal of the online retailer is to minimize the expected fulfillment costs, while satisfying the inventory constraints. Diverging from the approach of Jasin and Sinha (2015), we consider a different offline formulation of the problem. In our proposed model, the platform sequentially chooses "methods" for fulfilling incoming orders (which may contain one or more items). A method consists of a combination of item-facility pairs that specify from which warehouses the items will be shipped, determining therefore

whether multi-item orders items will be split or not. Within this framework, we develop a class of dynamic policies for order fulfillment that integrates techniques from randomized fulfillment, prophet inequalities, and subgradient methods. The most important contribution of this work is to explain how prophet inequality techniques can be used in the context of inventory management in order to derive algorithms that are asymptotically optimal and have strong approximation guarantees in non-asymptotic settings. Our findings reveal a straightforward and nearly optimal solution for fulfilling orders, provided that the online retailer has sufficient inventory, regardless of other problem specific parameters. Finally, our approach also uncovers new and simple asymptotically optimal policies for network revenue management (NRM).

Transitioning to Chapter 3, the dissertation focuses on an extension of our model in which the online retailer can use resources multiple times to fulfill orders. This extension captures scenarios such as rentals and finds relevance in various applications such as sports equipment rentals (like scuba diving, skiing/snowboarding, or climbing gear), photography equipment rentals, or party and event supplies rentals (like rentals of tables, chairs, decorations, and audio-visual equipment). Techniques used to prove our results are similar to the one adopted in Chapter 2, but more refined to accommodate the more intricate offline formulation which now requires that, at any time t, the amount of inventory in use does not exceed its capacity.

The research in these two chapters was conducted under the supervision of Ali Makhdoumi and Yehua Wei and the contributions are detailed in Amil et al. (2022).

1.2 Online Recommendation Systems

In Chapter 4, we study the complex challenge of online recommendation systems, where the goal is to tailor product offerings to meet customer needs. Recognizing the dynamic nature of users' preferences, we adopt a contextual bandit approach to model the problem, enabling the platform to adapt to the evolving preferences

over time and optimize their recommendations. Central to our model is the concept of regret minimization. We define regret as the difference in performance between the platform's recommendations and those of a hypothetical clairvoyant entity with full foresight of users' preferences. Our research rigorously formulates the problem as a contextual bandit model, accommodating an arbitrary sequence of customer valuations and a broad class of policies without imposing restrictive assumptions on the mappings from user contexts to product values.

We explore two distinct feedback settings: full-feedback, where the platform learns the user's valuation for each product feature post-recommendation, and bandit-feedback, where only the aggregate valuation for the recommended product is revealed to the platform. For both settings, we introduce online algorithms designed to adapt and learn from the feedback. Our analysis provides explicit regret bounds for these algorithms, illustrating their effectiveness as a function of the class of policies. Specifically, our regret depends on the class of mappings from contexts to actions, and exhibits the approximation versus statistical error trade-off. We then apply this framework to two specific classes of policies: finite mappings from contexts to actions and linear mappings from contexts to values. In these two examples, our results show that, in the full-feedback setting, the regret scales as $O(\sqrt{n})$, where n is the number of interactions with the customers. The $O(\sqrt{n})$ scaling indicates that as the number of interactions increases, the total regret grows proportionally to the square root of n. Thus, this slower growth rate implies that the platform's recommendations are becoming more effective over time, progressively narrowing the performance gap with the clairvoyant benchmark. Essentially, the platform learns from the full-feedback effectively, making fewer sub-optimal recommendations as it gains more experience. Conversely, in the bandit-feedback setting, the regret scales as $O(n^{2/3})$, reflecting the inherent challenges of learning under limited feedback. These examples demonstrate the versatility of our approach, showing its applicability to a

wide range of scenarios. More importantly, our algorithms work without the need to know the intricate processes by which customers develop their product preferences.

The research in this chapter was conducted under the supervision of Ali Makhdoumi during my second year at Duke.

2

Multi-Item Order Fulfillment

2.1 Introduction

Since the boom of e-commerce in the mid-90s, online retailing has grown into a multi-billion dollar industry. For example, in 2023, retail e-commerce sales amounted to approximately $5.8 trillion worldwide (Statista Research Department, 2023, 2024). Moreover, the increased adoption of the Internet across the globe and the continued growth of the industry suggest that it is likely that the demand for e-commerce will grow. In fact, e-commerce revenues are predicted to rise to more than $8 trillion worldwide by the end of 2027 (Statista Research Department, 2023, 2024). As the industry grows, some of the largest e-commerce platforms, such as Amazon, Alibaba, JD.com, and Walmart, are serving an increasing number of online customers worldwide. In order to address this growing customer base and stay competitive, e-commerce companies have to make complex real-time decisions to optimize their revenues and the customer experience. For example, an effective retail fulfillment strategy is crucial for e-commerce businesses. Indeed, unlike brick-and-mortar retailers, where the centralization of the fulfillment process streamlines the business to one place, online

retailers have more flexibility regarding the fulfillment of orders. Specifically, upon receiving an order, an online retailer has to decide from a complex set of alternatives, including which facility the items will ship from, by what shipping option, and whether or not multi-item orders will be split. This large set of alternatives makes it harder for e-commerce companies to make effective real-time fulfillment decisions and manage their distribution network. Traditionally, e-commerce companies adopt myopic policies to satisfy the demand (see, e.g., Xu et al., 2009), that is, they fulfill the orders in the cheapest way possible without considering future costs. However, by not accounting for future orders because of the limited inventory, they might lose the opportunity to maximize their profits (Acimovic and Graves, 2015).

2.1.1 Contributions and Main Approach

Due to the curse of dimensionality, finding the optimal dynamic policy for the online fulfillment problem is difficult, even if we assume that every order contains a single item (see, e.g., Xu et al., 2009; Acimovic and Graves, 2015; DeValve et al., 2023). As a result, the online fulfillment literature has focused on studying simple algorithms through asymptotic or competitive analysis (see, e.g., Acimovic and Farias, 2019). Jasin and Sinha (2015) are the first to study the general online multi-item fulfillment problem under this line of research. In particular, the authors design an innovative correlated rounding scheme that takes the solution of a deterministic linear program (LP) to construct a probabilistic fulfillment policy. For the rounding scheme, they derive a competitive ratio in the so-called "fluid scaling" regime in which both time and inventories are scaled to infinity. While the competitive ratio of their rounding scheme is not asymptotically optimal, Jasin and Sinha (2015) observe that the scheme is effective in numerical studies.

In this chapter, we revisit the work of Jasin and Sinha (2015) by considering a different offline formulation of the problem. Specifically, we combine ideas from

randomized fulfillment, prophet inequality, and projected supergradient method and propose a new class of computationally effective fulfillment policies with non-asymptotic (finite) competitive ratios, which are also asymptotically optimal when the inventory is scaled to infinity. For example, we derive a policy with a (non-asymptotic) average-case competitive ratio of $1 + (\kappa - 1)|q_{\max}|/\sqrt{s+3}$, where $q_{\max}$ is the largest possible order (in terms of variety of items), s is the minimum inventory available for any item, and κ is the maximum ratio between the cost of not fulfilling an order and the cost of using any other method. This result has two important implications. First, our policy is asymptotically optimal in the "fluid scaling" setting where both the inventory and time are scaled to infinity. Second, our competitive ratio is independent of the number of items and order types. This independence is important as, in practice, the number of items and order types are often very large, as big online retailers hold millions to hundreds of millions of item types in their facilities. In Table 2.1, we summarize a comparison of our result with the existing works in the multi-item order fulfillment literature. Note that, in this chapter, we propose a whole class of fulfillment policies, and the result presented in the next table is obtained by applying the closed-form prophet inequality of Alaei (2014). This competitive ratio can be improved using the tighter prophet inequality derived in Jiang et al. (2021).

There are a few points worth mentioning. First, in this chapter, instead of the item-facility based offline LP model proposed by Jasin and Sinha (2015), we consider an offline LP model that is method-based. This means that at each time period, the online retailer chooses a fulfillment method. While our method-based model has a large number of decision variables and constraints, we develop an effective supergradient method that solves a Lagrangian relaxation of a large LP (corresponding to the offline problem). This Lagrangian relaxation naturally decomposes the LP formulation, making the policy computationally viable. We further show that our supergradient method also leads to a policy that is asymptotically $O(\log |q_{\max}|)$-

competitive (as $s \to \infty$) for the general multi-item fulfillment framework of Jasin and Sinha (2015), answering a question raised by the authors. We note that this result is also independently resolved by Ma (2023), where the author took a different approach by building a novel virtual system with facilities opening according to Poisson processes, and items viewing the openings of facilities in their own "dilated" opening times. In contrast, our method relies on two ingredients: a Lagrangian relaxation and a sequential randomized rounding algorithm for the relaxed decomposed problems.

Table 2.1: Comparison of competitive ratios of multi-item order fulfillment papers.

	Multi-item Setting	Competitive Ratio	Asymptotic Optimality
This chapter	General	$1 + (\kappa - 1)\lvert q_{\max}\rvert/\sqrt{s+3}$	Yes
Jasin and Sinha (2015)	General	$\mathbb{E}_F[B(\lvert\mathcal{Q}\rvert)]$[1]	No
Ma (2023)	General	$\log(q_{\max}) + 1$	No
Zhao et al. (2020)	One RDC and one FDC[2]	2[3]	Yes[4]

We note that even though our method-based policy has a better competitive ratio

[1] Only applies to the asymptotic setting (in which the time horizon and inventory are scaled at the same rate). B is a function defined as $B(n) = (n+2)/4$ if n is even, $B(n) = (n+1)^2/4n$ if n is odd and F is the distribution defined by the proportion of total fixed costs incurred to fulfill an order type q (coming from region j) from facility k. Finally, $\mathcal{Q}$ denotes the random variable representing the arriving order type at one time period.

[2] RDC stands for "regional distribution center" and FDC stands for "front distribution center".

[3] Under the assumption that RDC has smaller fixed costs, but higher variable costs. The competitive ratio also holds under adversarial arrivals.

[4] The algorithm that achieves asymptotic optimality is different from the algorithm that achieves a competitive ratio of two.

compared to item-facility-based policies, it comes with a higher computational cost for solving an offline LP problem. Therefore, we view our work as complementary to those that study item-facility-based approaches.

We conclude this section by noting that our class of fulfillment policies provides a novel strategy for the general multi-item fulfillment problem. In particular, in addition to the randomized fulfillment, we add an accept/reject step to better control the on-hand inventory. This additional step allows us to leverage prophet inequalities and derive strong non-asymptotic guarantees. For example, we propose one such policy based on the single dimensional magician's problem of Alaei (2014) by creating a collection of magicians for all item-facility pairs and using this collection to provide an accept/reject strategy. It is important to note that our procedure does not require Alaei (2014)'s result. For example, we can alternatively use k-unit OCRS as a subroutine, and our result can be improved using the tight bound proved by Jiang et al. (2021). However, for simplicity of exposition, we adopt the closed-form guarantee from Alaei (2014). Finally, a special case of our algorithm also provides new asymptotically optimal bounds for network revenue management (NRM) problems (see, e.g., Ma et al., 2020; Baek and Ma, 2022) where the focus is on accept/reject decisions about the available resources (see Section 2.6 for further details).

2.1.2 Outline

We review the related literature in Section 2.1.3, formalize our e-commerce model in Section 2.2, and discuss our modeling assumptions in Section 2.2.1. In Section 2.3, we introduce the benchmark that we use in order to evaluate the performance of an algorithm and also provide our fulfillment strategy. In particular, in Section 2.3.2, we present our multi-item fulfillment strategy as a two-step procedure, combining probabilistic fulfillment and prophet inequality ideas. In Section 2.4, we discuss the analysis of an important component of our fulfillment strategy, which we call the

method-acceptance problem (see Section 2.3.2 for details). Finally, in Section 2.5, because the probabilistic fulfillment component of our strategy requires access to the solution of a large LP, we discuss how to obtain an approximate solution effectively. Proofs of selected results are presented in the main body of the chapter, while the remaining proofs are provided at the end of the chapter in Section 2.13.

2.1.3 Related Literature

Our work closely relates to three streams of literature: e-commerce fulfillment, prophet inequality, and dynamic stochastic optimization. For e-commerce fulfillment, the early work of Xu et al. (2009) studies an online multi-item order fulfillment model, analyzing the impact of periodically re-evaluating the real-time decision of assigning the arriving order to one or more warehouses under a myopic policy (without considering future orders). Related work by Acimovic and Graves (2015), however, shows a simple "CD - Textbook" example that illustrates how these types of myopic policies do not perform well even in simple settings, casting evidence for the importance of adopting forward-looking fulfillment policies. Following these papers, substantial research has been done in the e-commerce fulfillment literature. Acimovic and Graves (2017) explore how to use inventory replenishment as a way to alleviate the additional costs caused by demand spillover. Lei et al. (2018) and Harsha et al. (2019) study joint pricing and order fulfillment problems. Arlotto et al. (2023) study the impact of initial inventory placement to the regret of fulfillment policies. For a comprehensive tutorial on the fulfillment optimization problem, we refer Acimovic and Farias (2019) to interested readers. Very recently, researchers have also examined the benefits of limited flexibility of fulfillment networks in the single-item setting: Asadpour et al. (2020) and Xu et al. (2020) study the case when the unit reward is uniform through regret analysis, while DeValve et al. (2023) study a network with local and spillover fulfillment costs across distribution centers through asymptotic analysis and

simulations.

The paper by Jasin and Sinha (2015), which is the most relevant to our work, presents a comprehensive framework for online multi-item fulfillment. In particular, the authors design a correlated rounding scheme using the solution of a deterministic LP to construct a probabilistic fulfillment policy and provide an upper bound on its asymptotic competitive ratio. Their bound, however, is not asymptotically optimal (i.e., it does not go to 1 as inventory grows). By contrast, we provide a non-asymptotic analysis of the competitive ratio that only depends on the amount of available inventory. We note that different non-asymptotic analyses for multi-item fulfillment problems were studied by Zhao et al. (2020) with two distribution centers and by Andrews et al. (2019) with adversarial demand.

Our work also relates to the literature on prophet inequality and magician's problem. Prophet inequality is an online stochastic decision-making problem first studied by Krengel and Sucheston (1978) and Samuel-Cahn (1984) and further developed in Babaioff et al. (2007), Kleinberg and Weinberg (2012), Azar et al. (2014), and Dutting et al. (2020), among others. In the basic version of this problem, there are n random variables $X_1, X_2, \ldots, X_n$ with known distributions, but unknown realizations. These realizations are revealed sequentially, and the decision-maker (DM) wants to design a strategy (which is a stopping rule) that, upon observing the realization X_i (and all the values before it), decides either to choose i, stop, and get a reward X_i; or pass and move on to the next item (the DM is not allowed to come back to i ever again). The DM's goal is to maximize the expected reward. Krengel and Sucheston (1978), among others, present a strategy with expected reward $\frac{1}{2}\mathbb{E}\left[\max_{i=1,\ldots,n} X_i\right]$. The magician problem can be thought of as an extension of the prophet inequality, in which the decision-maker wants to choose (up to) k rewards. This fundamentally changes the problem because it is no longer a stopping time problem: the decision-maker needs to decide whether to collect each reward based on

its value and the number of collected rewards. Alaei (2014) presents an algorithm that guarantees a minimum ex-ante probability (at time 0) of $1 - 1/\sqrt{k+3}$ for collecting each reward and, therefore, achieves $1 - 1/\sqrt{k+3}$ of the offline benchmark (see also Jiang et al. (2021) for an analysis of the tightness of this bound). Our problem is different from both the prophet inequality and the magician problem and can be thought of as a multidimensional extension of the magician problem (also see Correa et al. (2019) for a survey on prophet inequality).

In general, our model can be viewed as a variant of dynamic matching problems. In the literature, online bipartite matching is studied, among others, in Feldman et al. (2009); Manshadi et al. (2012), k-stage variants of the classic vertex weighted bipartite b-matching is studied in Feng and Niazadeh (2020); dynamic matching problems in non-bipartite graphs are studied in Ashlagi et al. (2019a) and Ashlagi et al. (2019b); and the study of bipartite graphs where both sides arrive/leave over time is studied in Johari et al. (2021); Aouad and Saritaç (2020); Truong and Wang (2019); Castro et al. (2020). In addition, connections to assortment optimization are studied in Golrezaei et al. (2014); Ma and Simchi-Levi (2020); Aouad and Saban (2020); Feng et al. (2022); Désir et al. (2022); the dynamic matching with limited supply is studied in Elmachtoub and Levi (2016); Ma et al. (2021); the dynamic matching with returning suppliers is studied in Manshadi and Rodilitz (2020); Lo et al. (2020); Manshadi et al. (2022); information relaxation to design simple policies with guaranteed performances for a general class of stochastic dynamic optimizations is studied in Balseiro and Brown (2019); and joint inventory selection and online resource allocation Chen et al. (2022).

More broadly, our work is related to the literature on dynamic stochastic optimization. Specifically, our multi-item fulfillment problem complements the literature on dynamic resource allocation, which, in the past few years, has witnessed significant advancements, particularly in the realms of dynamic and stochastic knapsack

problems and dynamic matching. Recent contributions, such as the work by Arlotto and Gurvich (2019) on uniformly bounded regret in the multisecretary problem, and Aveklouris et al. (2021) on matching impatient and heterogeneous demand and supply, illustrate the progress achieved towards optimizing resource allocation in uncertain and evolving environments. Moreover, Balseiro et al. (2023)'s comprehensive survey cohesively synthesizes the diverse models and analyses the so-called dynamic resource-constrained reward collection problems. These studies not only underscore the theoretical depth and practical relevance of dynamic resource allocation but also pave the way for novel methodologies, such as primal-dual policies. For example, Wei et al. (2023) demonstrates the effectiveness of primal-dual policies in dynamic resource allocation, setting a new benchmark for future research. Such explorations are crucial for designing systems that are capable of adapting to the complexities of real-world scenarios, from network revenue management to online labor markets.

2.2 Model

We consider a setting in which orders arrive sequentially to an online retailer (throughout, we use the terms online retailer and platform interchangeably). The platform and the customers interact over a period of length T. During this period, at each time $t \in [T] := \{1, \ldots, T\}$, at most one order arrives. Upon the arrival of an order, the platform needs to decide the method for fulfilling this order by using items from one (or more) of its facilities (we also refer to facilities as warehouses). We allow the online retailer to not fulfill an arriving order, incurring a cost higher than fulfilling it with any other method. A fulfilling method consists of a set of facilities that will determine which warehouses the items will ship from and, in particular, whether multi-item orders will be split and shipped from different warehouses. The platform's goal is to minimize the sum of the fulfillment costs, that is, the cost incurred from fulfilling the orders arriving sequentially over the entire time horizon $[T]$.

Throughout this chapter, we let I be the set of all item types (indexed by i) and K be the set of facilities (indexed by k). We also let Q be the set of order types (indexed by q), where each element encodes characteristics related to a particular order, such as its item composition and the location from which the order is placed. To simplify our analysis, we assume the number of requests for each item in an order of type q is exactly one. This assumption is common in the multi-item literature, and it has been noted in Xu et al. (2009); Acimovic and Graves (2015) that orders with multiple requests for the same item are rare in practice. In addition, we slightly abuse the notation by using $|q|$ to represent the number of different items included in an order of type q and $i \in q$ to indicate that item $i \in I$ belongs to the order type $q \in Q$.

We let $p_t(q)$ denote the arrival probability of order type q at time t for all $t \in [T]$ and $q \in Q$, and we assume that the probabilities $p_t(\cdot)$ are independent across time. Note that we assume independence across time but the order type distribution can be non-stationary (hence, the subscript t). When an order type q arrives at time t, we consider the possible ways, or methods, available to the platform to fulfill the order, with the possibility of not fulfilling the requests for some items in q. For ease of notation, we assume, without loss of generality, that each method m can be used to fulfill only one order type, and we use $m \sim q$ to denote that method m is used for fulfilling order type q. Thus, each order type q can be fulfilled by multiple methods, but each method m is used to fulfill a unique order type. We let c_m denote the cost the platform incurs when it uses method m to fulfill the corresponding order type, and let M denote the set of all methods. For a method m such that $m \sim q$, we use $(i, k) \in m$ to represent that item i from facility k is being used for fulfilling order type q under method m. Moreover, we consider any sub-method $m' \subseteq m$ of m to be a method itself. Note that the set of items shipped using method m is always a subset

of the items requested by order type q, i.e.,

$$\{i \in I : \text{ there exists a unique } k \in K \text{ such that } (i,k) \in m\} \subseteq \{i \in I : \ i \in q\}\,.$$

The platform's cost is the sum of all costs incurred during the time horizon $[T]$. In our model, every facility $k \in K$ has a fixed inventory $S_{ik} \geqslant 0$ of units of item $i \in I$ and no inventory replenishment takes place during the time horizon $[T]$. Without loss of generality, we assume that $S_{ik} \geqslant s$ for all item-facility pairs for some $s > 0$.

2.2.1 Discussion of the Modeling Assumptions

We assume that the platform knows the distribution of arriving orders, i.e., $p_t(q)$ for $t \in [T]$ and $q \in Q$. This is a common assumption in online stochastic optimization (see, e.g., Mehta et al. (2013)) and it is reasonable because, in practice, large e-commerce platforms can use historical data to obtain a good estimate of the order distribution. To ensure feasibility, we assume that facility 0 contains an infinite amount of inventory for every item $i \in I$, i.e., $S_{i0} = \infty$ for each i. One can think of facility 0 as a dummy facility such that fulfilling an order by using this facility can be interpreted as not fulfilling that order. We let $m'(q)$ denote the corresponding "discard" method, i.e., the method in which the dummy facility is used to satisfy all items in q. Moreover, without loss of generality, we assume that for some $\kappa > 1$, $c_{m'(q)}/c_m \leqslant \kappa$, for all $m \sim q$ and any $q \in Q$, i.e., the ratio between the cost of not fulfilling an order and the cost of fulfilling it (with any other method) is bounded. We note that in practice the lost sale, as also noted in Jasin and Sinha (2015), Zhao et al. (2020), and DeValve et al. (2023), is around twice the maximum single-item cost. In our simulation studies, in Section 2.11, we use this fact to provide estimates for κ.

We next define a benchmark that we use to evaluate the performance of different fulfillment policies and provide a brief overview of our proposed strategy.

2.3 Offline Formulation and Fulfillment Strategy

In this section, as a benchmark, we consider a setting in which the platform knows all order types in advance and describe an offline optimization problem that formulates the corresponding platform's cost. Given our definition of method (see Section 2.2 for details), we let $x_m^t \in \{0,1\}$ denote the binary variable representing the offline decision of using method m to fulfill its corresponding order type, i.e., $x_m^t = 1$ if the platform uses method m and $x_m^t = 0$, otherwise. We denote by

$$\mathbf{x}^t = \{x_m^t\}_{m \in M},$$

the fulfillment decision at time t, and by $\mathbf{x} = \{\mathbf{x}^t\}_{t \in [T]}$ an offline fulfillment strategy over the entire time horizon $[T]$.

We next introduce some notations that we use in formulating the offline problem. Let $\mathcal{D}^{qt}$ be the number of arrivals of order type q at time t. Because of our assumption that at most one order arrives at each time t, observe that $\mathcal{D}^{qt}$ is a Bernoulli random variable with success probability $p_t(q)$. Now, since in our benchmark offline problem all order arrivals $\mathcal{D} = \{\mathcal{D}^{qt}\}_{q \in Q, t \in [T]}$ are known in advance, in order to minimize the platform's cost over the entire time horizon, we need to solve the following integer program (IP):

$$\text{OPT}(\mathcal{D}) \;:=\; \min_{\mathbf{x}} \sum_{t \in [T]} \sum_{q \in Q} \sum_{m : m \sim q} c_m x_m^t \tag{2.1}$$

$$\text{s.t.} \sum_{m : m \sim q} x_m^t = \mathcal{D}^{qt}, \;\; \forall q \in Q, t \in [T], \tag{2.2}$$

$$\sum_{t \in [T]} \sum_{m : (i,k) \in m} x_m^t \leqslant S_{ik}, \;\; \forall i \in I, k \in K, \tag{2.3}$$

$$x_m^t \in \{0,1\}, \;\; \forall m \in M, t \in [T]. \tag{2.4}$$

The objective is the platform's cost, which is the sum of the incurred costs from all periods. Moreover, the set of constraints (2.2) ensures that we fulfill order type q by

using exactly one method at time t (in the case in which $\mathcal{D}^{qt} = 1$). Note that this includes the "discard" method, i.e., fulfilling the order from the dummy facility. The set of constraints (2.3) ensures that, over the entire time horizon $[T]$, we do not fulfill orders using more than S_{ik} units of item $i \in I$ from facility $k \in K$. Finally, constraints (2.4) come from the platform's decision regarding using a method or not.

It is important to note that the optimization formulation of our offline problem differs from the one used to design fulfillment policies in Jasin and Sinha (2015). Specifically, in the formulation of Jasin and Sinha (2015), the number of decision variables scales polynomially in the number of order types and items in the orders. In our formulation, instead, the number of decision variables may increase exponentially with the number of items in the orders. However, even though their formulation is more compact, their LP relaxation does not lead to an asymptotically optimal policy. In contrast, as we will establish later, the optimal solution of the LP relaxation of our formulation leads to a policy that is both asymptotically optimal and guaranteed to have strong approximation factors in finite settings. In addition, while the number of decision variables in the LP relaxation of our formulation scales exponentially in the number of items in the orders, we develop an effective supergradient method to solve the corresponding LP relaxation approximately (within any factor) and show that the approximation factor directly carries over to the performance measure of our policy.

2.3.1 Competitive Ratio

Before describing the performance measure that we consider in this chapter, we formally introduce the definition of algorithm for an online fulfillment strategy.

Definition 2.1 (algorithm)**.** *An algorithm, denoted by* ALG*, at each time, specifies the method for fulfilling any arriving order type. More specifically, an algorithm specifies a collection of functions* $\{f^t\}_{t\in[T]}$ *(adapted to the natural filtration), where* f^t *is a mapping from the history of the interactions between the platform and the*

customers (and therefore, the available inventory levels) to a fulfillment decision $\mathbf{x}^t$ *at time* t*. For a given order arrival* $\mathcal{D}$*, we let* $\mathrm{ALG}(\mathcal{D})$ *be the random variable indicating the platform's cost when it adopts algorithm* ALG*.*

For a given algorithm, we define the following performance measure.

Definition 2.2 (competitive ratio)**.** *Given an algorithm ALG for fulfillment, we define the competitive ratio as*

$$\frac{\mathbb{E}\left[ALG(\mathcal{D})\right]}{\mathbb{E}\left[OPT(\mathcal{D})\right]}, \tag{2.5}$$

where $\mathbb{E}\left[ALG(\mathcal{D})\right]$ *is the expected cost obtained by algorithm ALG, compared against* $\mathbb{E}\left[OPT(\mathcal{D})\right]$ *which is the expected cost obtained by the optimal offline solution of* (2.1)*. The expectation is with respect to the arrival process* $\mathcal{D}$ *and the (possible) randomization in the algorithm.*

Note that computing $\mathrm{OPT}(\mathcal{D})$ requires solving problem (2.1), which is an integer program. Moreover, the integrality gap of this problem is not one, i.e., the optimal objective of the relaxed linear program and the integer program are not equal. We illustrate this through the following example.

Example 2.1. *Consider the following simple example. Suppose the platform has only one warehouse with inventory* $(1, 1, 1)$*, i.e., three items, each with inventory one. Now, consider the following sequence of three orders* $(1, 1, 0), (1, 0, 1)$*, and* $(0, 1, 1)$*, all coming from the same location. For each order* q*, let method* m_q^f *be the method that fulfills the entire order* q *from the warehouse with a cost of* 1*; method* m_q' *be the "discard" method for* q *(the method where the dummy facility satisfies all items in* q*), with a cost of* 10*; and let all other methods also have a cost of* 10*. Then, it is easy to see that the optimal solution in the relaxed problem of* (2.1) *is to use* $1/2$ *of* m_q^f *and* $1/2$ *of* m_q' *for each* q*, with a total cost of* $1.5 + 15 = 16.5$*. However, the optimal*

integer solution is to simply use m_q^f *for* $q = (1, 1, 0)$*, and use* m_q' *for orders* $(1, 0, 1)$ *and* $(0, 1, 1)$*, with a total cost of* $1 + 20 = 21$*.*

In general, computing $\mathrm{OPT}(\mathcal{D})$ even for one instance of $\mathcal{D}$ is NP-hard (see Jasin and Sinha (2015) for a further discussion on the hardness of this problem), which makes computing the exact value of $\mathbb{E}[\mathrm{OPT}(\mathcal{D})]$ difficult. Therefore, in this chapter, we focus on a lower bound of $\mathbb{E}[\mathrm{OPT}(\mathcal{D})]$ and compare the performance of our algorithm with this smaller benchmark. This lower bound is provided by the expected relaxation of (2.1) defined as follows.

$$\mathrm{OPT}^e \coloneqq \min_{\mathbf{x}} \sum_{t\in[T]} \sum_{q\in Q} \sum_{m:m\sim q} c_m x_m^t \tag{2.6}$$

$$\text{s.t.} \sum_{m:m\sim q} x_m^t = p_t(q),\ \forall q \in Q, t \in [T], \tag{2.7}$$

$$\sum_{t\in[T]} \sum_{m:(i,k)\in m} x_m^t \leqslant S_{ik},\ \forall i \in I, k \in K, \tag{2.8}$$

$$x_m^t \in [0, 1],\ \forall m \in M, t \in [T]. \tag{2.9}$$

Note that problem (2.6) differs from problem (2.1) in two ways. First, the right-hand side of the constraint (2.2) is replaced by its expected value. Second, the integer constraints (2.4) are replaced by their relaxed versions. Now, we perform a simple and standard aggregation across time periods to reduce the number of variables in (2.6) by defining

$$x_m \coloneqq \sum_{t\in[T]} x_m^t \text{ and } p^a(q) \coloneqq \sum_{t\in[T]} p_t(q).$$

Using $\langle \cdot, \cdot \rangle$ to denote the Euclidean inner product, we reformulate (2.6) as the following

time-aggregated optimization

$$\text{OPT}^E := \min_{\mathbf{x}} \sum_{q \in Q} \langle \mathbf{c}, \mathbf{x}[q] \rangle \tag{2.10}$$

$$\text{s.t.} \sum_{m:m \sim q} x_m = p^a(q), \ \forall q \in Q, \tag{2.11}$$

$$\sum_{m:(i,k) \in m} x_m \leqslant S_{ik}, \ \forall i \in I, k \in K, \tag{2.12}$$

$$x_m \in [0, T], \ \forall m \in M, \tag{2.13}$$

where $\mathbf{c} = (c_m)_{m \in M}$ and $\mathbf{x}[q] = (x_m \mathbb{1}\{m \sim q\})_{m \in M}$ denotes the subvector of $\mathbf{x}$ corresponding to methods that can be used to fulfill order type q. Note that problem (2.6) and problem (2.10) achieve the same optimal value, i.e., $\text{OPT}^e = \text{OPT}^E$. More precisely, problem (2.6) and problem (2.10) have the same objective function and the same feasible set. To see this, note that, if $\{x_m^t\}_{m \in M, t \in [T]}$ satisfies (2.8), then it satisfies (2.12) (by definition of x_m); and if $\{x_m^t\}_{m \in M, t \in [T]}$ satisfies (2.7), then by summing over t, we obtain (2.11). Thus, we have $\text{OPT}^E \leqslant \text{OPT}^e$. Suppose now that $\{x_m\}_{m \in M}$ satisfies (2.11) and (2.12). Then,

$$x_m^t = \frac{x_m p_t(q)}{p^a(q)} \in [0, 1]$$

is feasible for problem (2.6). Indeed, if $\sum_{m:m \sim q} x_m = p^a(q)$, then

$$\sum_{m:m \sim q} x_m^t = \sum_{m:m \sim q} \frac{x_m p_t(q)}{p^a(q)} = p_t(q),$$

and, if $\sum_{m:(i,k) \in m} x_m \leqslant S_{ik}$, then

$$\sum_{t \in [T]} \sum_{m:(i,k) \in m} x_m^t = \sum_{t \in [T]} \sum_{m:(i,k) \in m} \frac{x_m p_t(q)}{p^a(q)} = \sum_{m:(i,k) \in m} \frac{x_m p^a(q)}{p^a(q)} \leqslant S_{ik}.$$

Thus, we obtain $\text{OPT}^e \leqslant \text{OPT}^E$. Given this equivalence, note that if $\{y_m\}_{m \in M}$ is a

solution to (2.10), then we can obtain a solution to (2.6) as follows

$$y_m^t = \frac{y_m p_t(q)}{p^a(q)}. \tag{2.14}$$

We next establish that the objective of problem (2.10) is indeed a lower bound on the expected value of the objective of problem (2.1).

Lemma 2.1. *The objective of problem* (2.10) *is weakly smaller than the expected value of the objective of problem* (2.1)*, i.e.,*

$$OPT^E \leqslant \mathbb{E}\left[OPT(\mathcal{D})\right]. \tag{2.15}$$

As we will see later, the optimal solution to (2.10) does not only provide an upper bound on the optimal solution of (2.1) but also plays a crucial role in designing our algorithm. Moreover, problem (2.10) is an LP, which is tractable when the number of methods $|M|$ (hence the number of variables) is small. In general, $|M|$ can grow exponentially with the number of items in the order types. Specifically, the number of decision variables in (2.10) corresponding to an order type q is $(|K|+1)^{|q|}$ (assuming that every facility has the same set of items). This further shows the combinatorial nature of our e-commerce problem. As a result, in many practical scenarios, problem (2.10) cannot be solved using off-the-shelf solvers, as it contains too many decision variables. In order to overcome this problem, we develop a computationally viable method for solving (2.10) approximately through a supergradient method for its dual problem (as (2.10) decomposes into many smaller LPs under a Lagrangian relaxation). We discuss this method in detail in Section 2.5. Next, we present a high-level idea of our fulfillment strategy and show how to use the solution of (2.10) to design an online algorithm for our multi-item order fulfillment framework.

Algorithm 1 Online algorithm for fulfillment

Offline Process:

- Solve the expected LP in (2.10) and obtain the corresponding $\mathbf{y} = \{y_m^t\}_{m \in M, t \in [T]}$ (through (2.14))

Online Process:

- **For** $t = 1, \ldots, T$:

  Let q be the order that arrives at time t and $M(q)$ be the set of all available fulfillment methods for order type q (including the "discard" method $m'(q)$).

  1. Draw a method with probability $y_m^t / p_t(q)$ for $m \in M(q)$.
  2. If method $m \in M(q)$ is drawn, then:
     (a) Decide whether to accept or reject method m. If accepted, use method m to fulfill q. If rejected, use the "discard" method $m'(q)$.

2.3.2 Fulfillment Policies

In this section, we provide an overview (in Algorithm 1) of our strategy, which consists of a two-step procedure: an offline process and an online process. The offline process starts at time 0 (before the fulfillment process starts) and consists in solving the LP relaxation (2.10) of the offline problem and obtaining the corresponding solution $\mathbf{y} = \{y_m^t\}_{m \in M, t \in [T]}$. Note that, as we have shown previously, we can obtain $\mathbf{y} = \{y_m^t\}_{m \in M, t \in [T]}$ using the relationship specified by (2.14). The relaxation of the offline problem essentially reformulates the fulfillment process as a deterministic process by leveraging the knowledge about the expected demand and satisfying the constraints in expectation. After solving (2.10) and obtaining $\mathbf{y} = \{y_m^t\}_{m \in M, t \in [T]}$, the solution can be interpreted as a sequence of probability distributions over the

set of fulfillment methods M. It is worth mentioning that the current literature on multi-item fulfillment focuses on providing a probability distribution over the set of item-facility pairs (see, e.g., Jasin and Sinha, 2015), i.e., the frequency with which each facility should be used to fulfill an item in a given order type. In contrast, our offline process directly provides a "guide" for the choice of the fulfillment method, which already prescribes a complete picture of how the order should be split among the available facilities. This key difference turns out to be crucial for designing policies with strong non-asymptotic performance guarantees that are also asymptotically optimal. Note that the LP relaxation (2.10) was also considered by Jasin and Sinha (2015), but the authors cautioned that such LP has too many variables and constraints to be solved to optimality. To address this problem, in Section 2.5, we demonstrate that a Lagrangian relaxation naturally decomposes the LP formulation. Based on this decomposition, we present a supergradient method that effectively finds near-optimal solutions, making the policy computationally viable, with a small sacrifice in the competitive ratio.

After solving the LP relaxation, our online process consists of two steps. First, we perform a probabilistic fulfillment step, i.e., when an order type arrives, we randomly draw a method based on the LP solution. Second, on top of randomization, our algorithm has an additional step in which we decide whether to accept or reject the randomly drawn method. When the first step is fixed, the second step can be viewed as a dynamic decision problem on its own, which we refer to as the method-acceptance problem. For the method-acceptance problem, at each time t, exactly one method arrives (where the arriving method is drawn from the fixed probabilistic procedure) and the decision maker has to immediately choose whether to use it based on the available inventories. If they decide to use/accept the method, then the inventories corresponding to the items in the order will be consumed (from the facilities specified by the method). If they decide to reject the method, then the online retailer uses

what we call the discard method, incurring the maximum cost possible (by redirecting the order to the dummy facility).

2.4 Analysis of the Method-Acceptance Problem

In this section, we propose and analyze a strategy for the method-acceptance problem. Note that finding the optimal policy for this problem is intractable due to the curse of dimensionality. As a result, instead of directly optimizing for a strategy that considers the trade-off between using resources to fulfill the current order and holding the inventories for the future, we focus on constructing a strategy that provably accepts any method at any period t with high probability. Intuitively, such a strategy would be close to optimal, as its expected cost will be close to the objective of the expected LP in (2.10), which is a lower bound for the expected cost of any fulfillment strategy. Next, we formally introduce the definition of such strategy, which we call γ-conservative method-acceptance strategy.

Definition 2.3 (γ-conservative method-acceptance strategy)**.** *Consider a sequence of (possibly randomized) decision rules $\pi = \{\pi^t\}_{t\in[T]}$ that determine whether to accept a method based on the arriving method type and the history that occurred until time t. Specifically, $\pi^t : H^t \times M \to \{0, 1\}$, where H^t is the set of all possible histories until time t and M is the set of all method types. Then, we say that π is a γ-conservative method-acceptance strategy if for any $m \in M$ and $t \in [T]$, we have*

$$\mathbb{P}(\textit{accepting method at } t \mid \textit{the method is type } m) = \sum_{h^t \in H^t} \mathbb{P}(\pi^t(h^t, m) = 1) \cdot \mathbb{P}(h^t) \geqslant \gamma.$$

When each order (and, therefore, each method) consists of at most one item, then the γ-conservative method-acceptance strategy stated above reduces to the celebrated γ-conservative magician strategy studied by Alaei (2014). As a result, designing a γ-conservative method-acceptance strategy can be viewed as a multi-dimensional

extension of Alaei's magician strategy. Next, we show that a γ-conservative method-acceptance strategy directly leads to a non-asymptotic competitive ratio for our fulfillment policy.

Lemma 2.2. *For any online multi-item fulfillment problem, using a γ-conservative method-acceptance strategy, Algorithm 1 achieves a competitive ratio of at most*

$$1 + (\kappa - 1)(1 - \gamma),$$

where we recall that κ is the maximum ratio between the cost of not fulfilling an order and the cost of using any other method.

As we will show in Section 2.4.1, the question of finding strategies that provably accept arriving methods with probability at least γ naturally connects to the prophet inequality literature. Indeed, in this fulfillment context, prophet inequalities allow to design online strategies with two desirable features: these strategies are robust to arbitrary bad instances of order (or method) arrivals, and they satisfy a provable performance guarantee compared to the optimal solution in hindsight. Because of these two features, we also note that γ-conservative method-acceptance strategies naturally lead to γ-competitive policies for network revenue management problems, where multiple capacity-constrained resources are sold to a stream of arriving customers. We elaborate on this connection in Section 2.6.

2.4.1 Magician-Based Strategy for the Method-Acceptance Problem

In this section, we design a γ-conservative method-acceptance strategy that uses a set of conservative magician strategies of Alaei (2014) as a subroutine. The main idea is to construct a magician (i, k) for each item-facility pair, which controls the amount of inventory level S_{ik} of item i in a facility k. Specifically, at each time period t, for each (i, k) pair, magician (i, k) is faced with the problem of deciding whether

to make inventory (i, k) available for fulfillment. If magician (i, k) makes item i from facility k available for fulfillment, one unit of the corresponding inventory may be consumed. If they decide not to make their inventory available, then no resource is consumed. Magician (i, k) wants to ensure $\gamma_{ik} \in [0, 1]$ ex-ante probability (before any order realizes) of making their own inventory available for fulfillment at any time period without running out of it by the end of the time horizon $[T]$. Next, we formally define the problem for each magician (i, k).

Definition 2.4 (Magician (i, k) problem)**.** *For each (i, k) pair, where item $i \in I$ is in facility $k \in K$, imagine a game in which a magician has to manage the consumption of inventory (i, k) (over a time horizon $[T]$), with the goal of not running out of it by the end of the horizon. The magician starts with $S_{ik} > 0$ units of inventories of item i from facility k (with $S_{ik} < T$). Then, at each time $t \in [T]$, the magician decides whether to make one unit of inventory (i, k) available for fulfillment. If the magician chooses to make inventory (i, k) available for fulfillment, then with probability at most μ_{ik}^t, the inventory is consumed. Before making their decision at time t, the magician learns μ_{ik}^t and moreover, it is guaranteed that $\sum_{t=1}^{T} \mu_{ik}^t \leqslant S_{ik}$. Magician (i, k) would like to devise a γ_{ik}-conservative strategy, i.e., a strategy that guarantees an ex-ante (at time 0) probability of at least $\gamma_{ik} \in [0, 1]$ of making inventory (i, k) available for fulfillment at any time period t.*

In Algorithm 2, we present a method-acceptance strategy that leverages a set of γ_{ik}-conservative strategies (described in Definition 2.4) as subroutines. Next, we discuss the details of our method-acceptance strategy. For each item i in facility k, we define a magician problem with

$$\mu_{ik}^t := \sum_{m:(i,k)\in m} y_m^t,$$

where $\mathbf{y} = \{y_m^t\}_{m\in M, t\in[T]}$ is obtained through (2.14). Remember that, as described

Algorithm 2 A strategy for the method-acceptance problem

Inputs at $t = 0$:

- For each (i, k), the probabilities $\{\mu_{ik}^t\}_{t\in[T]}$, inventory S_{ik} and $\gamma_{ik} \in [0, 1]$ as in Definition 2.4.

For $t = 1, \ldots, T$:

- Let m be the method that is drawn at time t.
- If all magicians $(i, k) \in m$ make their inventory available for fulfillment, then accept method m. Else, reject m.

---

in Definition 2.4, μ_{ik}^t must be an upper bound on the probability of inventory (i, k) being consumed at time t. Because of our fulfillment strategy, this upper bound is provided by the sum of all probabilities $y_m^t \mathbb{1}\{(i, k) \in m\}$ of using a method that prescribes consuming (i, k) at time t. Note, moreover, that $\mu_{ik}^t \leqslant 1$ is ensured by the constraints in (2.7) and $\sum_{t=1}^T \mu_{ik}^t \leqslant S_{ik}$ holds by the constraints in (2.8).

At time 0 (before starting our fulfillment process), for each item-facility pair (i, k), we consider a magician problem as in Definition 2.4. For this problem, in Section 2.9, we present a threshold based γ_{ik}-conservative strategy for magician (i, k), inspired by Alaei (2014), with $\gamma_{ik} = 1 - 1/\sqrt{S_{ik} + 3}$. In particular, magician (i, k) adaptively computes a sequence of thresholds $\{\theta_{ik}^t\}_{t=1}^T$ and makes item i from facility k available for fulfillment at time t if its number of used inventories prior to time t is below this threshold. The thresholds are determined by the probabilities $\{\mu_{ik}^t\}_{t\in[T]}$ of consuming inventories throughout the fulfillment process (see Section 2.9 for details). The decision about whether to accept or reject the randomly drawn fulfillment method m is then specified by Algorithm 2, based on the "recommendations" provided (at time 0) by all magicians $(i, k) \in m$. Specifically, if all magicians decide to make their

own inventory available for fulfillment at time t, we accept method m for fulfillment. Otherwise, we reject method m and fulfill order type q from the dummy facility 0 (i.e., using the discard method $m'(q)$), incurring cost $c_{m'(q)}$. Next, we present the formal statement about our γ-conservative method-acceptance strategy with the exact γ. This result is obtained by using prophet inequality results from the literature and a union bound. For simplicity of exposition, we provide a γ-conservative method-acceptance strategy using the closed-form guarantee from Alaei (2014). We remark that our result can be improved using the analysis from Jiang et al. (2021), where the bound has more nuanced dependencies on the parameters of problem instances. In addition to the statement, we also include the formal proof of the result as it illustrates the connection between the prophet inequality and the multi-item fulfillment model through a union bound.

Proposition 2.1. *For any online multi-item fulfillment problem, when* $\gamma_{ik} = 1 - 1/\sqrt{S_{ik}+3}$ *for each pair* (i,k)*, Algorithm 2 provides a* γ*-conservative method-acceptance strategy with*

$$\gamma = 1 - \frac{|q_{\max}|}{\sqrt{s+3}},$$

where $|q_{\max}|$ *denotes the size of the largest possible order and* $s = \min_{i \in I, k \in K, S_{ik}>0} \{S_{ik}\}$.

Proof of Proposition 2.1: We want to prove that Algorithm 2 is a γ-conservative method-acceptance strategy, i.e.,

$$\mathbb{P}(\text{accepting the method at } t \mid \text{the method is of type } m) \geqslant \gamma,$$

with $\gamma = 1 - |q_{\max}|/\sqrt{s+3}$. Let us start by describing how our fulfillment setting relates to each magician (i,k) problem in Definition 2.4. First, remember that magician (i,k) in Definition 2.4 manages the consumption of inventory for item i in facility k during the time horizon $[T]$. Now, note that this problem directly relates to the original magician's problem in Alaei (2014). Indeed, the amount of inventory

S_{ik} can be thought of as the number of magic wands available to the magician; the decision about making the inventory available for fulfillment corresponds to the decision of opening a box; and the consumption of inventory coincides with a magic wand breaking in Alaei's problem. Because of this correspondence, if we adopt the magician's strategy from Alaei (2014) for our problem in Definition 2.4, we have that, whenever $\gamma_{ik} \leqslant 1 - 1/\sqrt{S_{ik}+3}$, magician (i,k) never requires more than S_{ik} inventory and they are guaranteed an ex-ante probability (before the fulfillment process starts) of at least γ_{ik} of making the inventory available at any time period t. Assuming that method type m is drawn at time t, note that each magician (i,k) makes the decision independently of m, while the method-acceptance policy in Algorithm 2 makes the decision based on the set of magicians $(i,k) \in m$ (which depends on m).

Formally, denoting by $A^t_{ik} = \{\text{inventory } (i,k) \text{ available for fulfillment at } t\}$ the event in which magician (i,k) makes inventory i from facility k available for fulfillment at time t, we have that, according to Algorithm 2, for any $m \in M$ and $t \in [T]$

$$\begin{aligned}
\mathbb{P}(\text{accepting the method at } t \mid \text{the method is of type } m) &\overset{(a)}{=} \mathbb{P}\left(\bigcap_{(i,k)\in m} A^t_{ik}\right) \\
&\overset{(b)}{\geqslant} 1 - \sum_{(i,k)\in m} (1 - \mathbb{P}(A^t_{ik})) \\
&\overset{(c)}{\geqslant} 1 - \sum_{(i,k)\in m} (1 - \gamma_{ik}) \\
&\overset{(d)}{\geqslant} 1 - \frac{|q_{\max}|}{\sqrt{s+3}},
\end{aligned}$$

where (a) follows from the definition of Algorithm 2, (b) holds by using union bound, (c) follows from (Alaei, 2014, Theorem 4) because Definition 2.4 is an instance of the magician's problem, and (d) holds by setting $\gamma_{ik} = 1 - 1/\sqrt{S_{ik}+3}$ and the fact that $S_{ik} \geqslant s$. This completes the proof. ■

Now, note that our fulfillment strategy specified in Algorithm 1, together with the γ-conservative method-acceptance strategy (specified in Algorithm 2) as a subroutine, provides the following performance guarantee for our multi-item fulfillment setting.

Theorem 2.1. *For any online multi-item fulfillment problem, when Algorithm 1 uses Algorithm 2 as a subroutine in Step 2 of the online process, we obtain a fulfillment strategy with a competitive ratio of at most*

$$1 + \frac{(\kappa - 1)|q_{\max}|}{\sqrt{s+3}}, \tag{2.16}$$

where $|q_{\max}|$ denotes the size of the largest possible order and $s = \min_{i \in I, k \in K, S_{ik} > 0} \{S_{ik}\}$.

Proof of Theorem 2.1: Let $\mathbb{E}\left[\mathrm{ALG}_1(\mathcal{D})\right]$ denote the expected cost incurred by Algorithm 1. Then, using the γ-conservative method-acceptance strategy defined in Algorithm 2 with $\gamma = 1 - |q_{\max}|/\sqrt{s+3}$, we have that under Algorithm 1

$$\frac{\mathbb{E}\left[\mathrm{ALG}_1(\mathcal{D})\right]}{\mathbb{E}\left[\mathrm{OPT}(\mathcal{D})\right]} \overset{(a)}{\leqslant} 1 + (\kappa - 1)(1 - \gamma) = 1 + \frac{(\kappa - 1)|q_{\max}|}{\sqrt{s+3}},$$

where (a) follows from Lemma 2.2. This completes the proof. ■

There are a few points worth mentioning. First, the competitive ratio of our fulfillment algorithm depends only on the largest possible order (in terms of variety of items), the minimum inventory available for any item, and κ. In particular, the bound does not depend on the number of items and order types. This independence is important as, in practice, the number of items and order types are often very large, as large e-retailers hold millions to hundreds of millions of item types in their facilities. In Section 2.12.1, we also show how our analysis can be extended to establish a bound that depends on the average order size rather than the maximum order size. Second, the offline computation of our algorithm involves solving problem (2.10), which, even though it is a linear program, may have too many decision variables. As we show

in the next section, however, we can solve this problem approximately through a supergradient method. We conclude this section by noting that our performance guarantee continues to hold for a slight variation of Algorithm 2, described below.

Remark 2.1. *In Algorithm 2, we accept a method m to fulfill i from k if all magicians $(i', k') \in m$ make their inventory available. Otherwise, we reject method m. Our performance guarantee established in Theorem 2.1 continues to hold if instead of rejecting method m we fulfill $(i, k) \in m$ whose magicians have made their inventory available. This slight variation of Algorithm 2 rejects fewer items and, in practice, can perform better.*

2.5 Approximately Solving the Offline Relaxation

In Algorithm 1, we need access to a solution y_m^t obtained through (2.14) after solving (2.10) for all $m \in M, t \in [T]$. Given the combinatorial nature of our e-commerce model, the number of variables in problem (2.10) scales with $O(|Q||K|^{|q_{\max}|})$, which can be extremely large for many practical scenarios. Motivated by this, we first establish a simple lemma that an approximate solution suffices for designing our algorithm (and that the approximation factor directly carries over to the competitive ratio of the algorithm). We then complement this observation by developing a supergradient method to approximately solve problem (2.10).

Definition 2.5 (ϵ-approximation)**.** *Let $\{y_m\}_{m\in M}$ be an optimal solution to the LP relaxation* (2.10)*. The variables $\{\hat{\mathbf{y}}_m\}_{m\in M}$ form an ϵ-approximation solution to* (2.10) *if they satisfy the constraints and*

$$\sum_{q\in Q}\sum_{m:m\sim q} c_m \hat{y}_m \leqslant (1+\epsilon)\sum_{q\in Q}\sum_{m:m\sim q} c_m y_m,$$

for some $\epsilon > 0$.

Given this definition, we next show that our algorithm also works with an ϵ-approximation of problem (2.10) (with a small sacrifice on the performance guarantee).

Lemma 2.3. *For any online multi-item fulfillment problem, if we use an ϵ-approximation of relaxation* (2.10) *in Algorithm 1, then the algorithm achieves a competitive ratio of at most*

$$(1+\epsilon)(1+(\kappa-1)(1-\gamma)).$$

2.5.1 A Supergradient Method to Solve Problem (2.10)

In this section, we develop an algorithm to obtain an ϵ-approximation solution to problem (2.10). First, note that problem (2.10) reduces the number of variables by a factor of T. However, it still has a large number of decision variables. This motivates us to develop a supergradient method based on a Lagrangian relaxation, which we define below.

Definition 2.6 (Lagrangian relaxation and dual)**.** *For $\boldsymbol{\lambda} = \{\lambda_{ik}\}_{i\in I,k\in K} \in \mathbb{R}_+^{|I||K|}$, define the Lagrangian function $L(\boldsymbol{\lambda},\mathbf{x}) \coloneqq \sum_{q\in Q}\langle \mathbf{c}, \mathbf{x}[q]\rangle + \sum_{i\in I,k\in K}\lambda_{ik}(\sum_{m:(i,k)\in m} x_m - S_{ik})$, and let*

$$L(\boldsymbol{\lambda}) \coloneqq \min_{\mathbf{x}} \; L(\boldsymbol{\lambda},\mathbf{x}) \tag{2.17}$$

$$s.t. \quad \sum_{m:m\sim q} x_m = p^a(q), \;\; \forall q \in Q,$$

$$x_m \in [0,T], \;\; \forall m \in M,$$

denote the Lagrangian relaxation of (2.10) *with parameter $\boldsymbol{\lambda}$. We also let $L^* \coloneqq \max_{\boldsymbol{\lambda}\in\mathbb{R}_+^{|I||K|}} L(\boldsymbol{\lambda})$ denote the Lagrangian dual of problem* (2.10).

Note that weak and strong duality directly imply that

$$L(\boldsymbol{\lambda}) \leqslant L^* = \mathrm{OPT}^E.$$

Given this relationship between the primal and the Lagrangian dual, we propose Algorithm 3, a (projected) supergradient method for the Lagrangian dual that generates an approximately optimal primal solution. Note that in (2.17) we need to compute $L(\boldsymbol{\lambda})$ for different values of $\boldsymbol{\lambda}$, i.e., we need to solve a minimization problem for each $\boldsymbol{\lambda}$. Moreover, $L(\boldsymbol{\lambda})$ is concave in $\boldsymbol{\lambda}$. This motivates us to use a supergradient method.

Now note that, although the number of variables in the minimization problem (2.17) is large, this problem becomes separable in q. Indeed, we can rewrite (2.17), after appropriate scaling of the decision variables, as

$$\begin{aligned} L(\boldsymbol{\lambda}) = \min_{\mathbf{x}} \ & \sum_{q \in Q} p^a(q) \left(\langle \mathbf{c}, \mathbf{x}[q] \rangle + \sum_{i \in q, k \in K} \lambda_{ik} \sum_{m:(i,k) \in m, m \sim q} x_m \right) - \sum_{i \in I, k \in K} \lambda_{ik} S_{ik} \quad (2.18) \\ \text{s.t.} \quad & \sum_{m: m \sim q} x_m = 1, \ \forall q \in Q, \\ & x_m \in [0,1], \ \forall m \in M, \end{aligned}$$

where the first term in the objective depends on $\mathbf{x}$ and the last term depends only on $\boldsymbol{\lambda}$. Next, for each $q \in Q$, define

$$\begin{aligned} L_q(\boldsymbol{\lambda}) \coloneqq \min_{\mathbf{x}[q]} \ & \langle \mathbf{c}, \mathbf{x}[q] \rangle + \sum_{i \in q, k \in K} \lambda_{ik} \left(\sum_{m:(i,k) \in m, m \sim q} x_m \right) \quad (2.19) \\ \text{s.t.} \quad & \sum_{m: m \sim q} x_m = 1, \\ & x_m \in [0,1], \ \forall m \in M \text{ such that } m \sim q. \end{aligned}$$

Then, we have

$$L(\boldsymbol{\lambda}) = \sum_{q \in Q} p^a(q) L_q(\boldsymbol{\lambda}) - \sum_{i \in I, k \in K} \lambda_{ik} S_{ik}. \quad (2.20)$$

Given the decomposition in (2.20), in order to find the optimal solution of (2.17), denoted by $\mathbf{x}^*(\boldsymbol{\lambda})$, it is enough to find an optimal solution of (2.19) for each $q \in Q$.

Moreover, note that now the number of decision variables in (2.19) is $(|K|+1)^{|q|}$ (for a given order type q). This is much smaller than $\sum_{q\in Q}(|K|+1)^{|q|}$, the number of decision variables in (2.17), as Q can be very large. Therefore, solving (2.19) for each individual q is considerably more efficient than solving (2.17) as one large LP.

Now, let $g_q(\boldsymbol{\lambda})$ be a supergradient of $L_q(\cdot)$ at $\boldsymbol{\lambda}$. Then, because of (2.20), a supergradient of $L(\boldsymbol{\lambda})$ at $\boldsymbol{\lambda}$ is given by

$$G(\boldsymbol{\lambda}) \coloneqq \sum_{q\in Q} p^a(q) g_q(\boldsymbol{\lambda}) - \boldsymbol{S},$$

where $\boldsymbol{S} = (S_{ik})_{i\in I, k\in K}$. We now state a simple and yet important lemma that finds a supergradient of $L_q(\cdot)$ at $\boldsymbol{\lambda}$.

Lemma 2.4. *Let $q \in Q$ be fixed and $\mathbf{x}^*[q](\boldsymbol{\lambda}) = (x^*_m(\boldsymbol{\lambda})\mathbb{1}\{m \sim q\})_{m\in M}$ be an optimal solution of $L_q(\boldsymbol{\lambda})$. Then, for $\boldsymbol{\lambda} \in \mathbb{R}_+^{|I||K|}$,*

$$g_q(\boldsymbol{\lambda}) = \left(\sum_{m:(i,k)\in m, m\sim q} x^*_m(\boldsymbol{\lambda}) \right)_{i\in I, k\in K}$$

is a supergradient of $L_q(\cdot)$ at $\boldsymbol{\lambda}$.

Given Lemma 2.4, we formally state our projected supergradient method as Algorithm 3. As it turns out, Algorithm 3 can find approximate solutions to both the primal and the (Lagrangian) dual formulations. Next, we formalize this statement through a pair of propositions.

Proposition 2.2. *Let C be a constant independent of J such that $\|G(\boldsymbol{\lambda}^{(j)})\|_2^2 \leqslant C \cdot OPT^E$ for all $j = 1, \ldots, J$, and let*

$$L^{(J)}_{\text{avg}} \coloneqq L\left(\frac{1}{J} \sum_{j=1}^{J} \boldsymbol{\lambda}^{(j)} \right).$$

Then, given a step-size α_J (as defined in Algorithm 3), at iteration J, we have

$$L^* - L^{(J)}_{\text{avg}} \leqslant \frac{\|\boldsymbol{\lambda}^*\|_2^2 + \alpha_J^2 J (C \cdot OPT^E)}{2\alpha_J J}. \tag{2.21}$$

Algorithm 3 Projected supergradient method with fixed step-size

Input: Initialize $\boldsymbol{\lambda}^{(1)} = \mathbf{0}$, with $\mathbf{0} \in \mathbb{R}_+^{|I||K|}$, $\alpha_J \in [0,1]$ such that $\alpha_J J \to \infty$ and $\alpha_J \to 0$ as $J \to \infty$.

For $j = 1, \ldots, J$:

1. For each $q \in Q$, solve $L_q(\boldsymbol{\lambda}^{(j)})$ in (2.19) to obtain an optimal $\mathbf{x}[q]^{(j)} := \mathbf{x}[q](\boldsymbol{\lambda}^{(j)})$.
2. Let $G(\boldsymbol{\lambda}^{(j)}) = \sum_{q \in Q} p^a(q) g_q(\boldsymbol{\lambda}^{(j)}) - \boldsymbol{S}$, $g_q(\boldsymbol{\lambda}^{(j)}) = \left(\sum_{m:(i,k)\in m, m\sim q} x_m^{(j)}\right)_{i\in I, k\in K}$;
3. Update $\boldsymbol{\lambda}^{(j+1)} = \max\{0, \boldsymbol{\lambda}^{(j)} + \alpha_J G(\boldsymbol{\lambda}^{(j)})\}$;

Output: $\bar{\mathbf{x}} := \frac{1}{J}\sum_{j=1}^{J}\left(\sum_{q\in Q} p^a(q)\mathbf{x}[q]^{(j)}\right)$.

Note that, as we show at the end of the proof of Proposition 2.2, the constant C always exists as $x^*(\lambda)$ is bounded for any λ. Now, using equation (2.21) in Proposition 2.2, we have that at iteration J of Algorithm 3

$$0 \leqslant L^* - L_{\text{avg}}^{(J)} \leqslant \frac{\|\boldsymbol{\lambda}^*\|_2^2 + \alpha_J^2 J(C \cdot OPT^E)}{2\alpha_J J},$$

implying that, when $\alpha_J J \to \infty$ and $\alpha_J \to 0$ as $J \to \infty$, $\lim_{J\to\infty} L_{\text{avg}}^{(J)} = L^*$. This shows the convergence of our supergradient method. Note that although the above proposition suggests that Algorithm 3 can find an approximately optimal dual solution (when, e.g., $\alpha_J = 1/\sqrt{J}$), we also need to find an optimal primal solution that is feasible with respect to the inventory constraint. Next, we show that under some mild assumptions, the average primal solution (up to iteration J) is guaranteed to be close to be optimal, and it satisfies the inventory constraint approximately. We formally state this in the next theorem.

Theorem 2.2. *Let C be a constant independent of J such that $\|G(\boldsymbol{\lambda}^{(j)})\|_2^2 \leqslant C \cdot OPT^E$ for all $j = 1, \ldots, J$. Denote by $\bar{\mathbf{x}} = \frac{1}{J}\sum_{j=1}^{J}\left(\sum_{q\in Q} p^a(q)\mathbf{x}[q]^{(j)}\right)$, where $\mathbf{x}[q]^{(j)}$ is*

defined as in Algorithm 3. Then, under Algorithm 3, we have that

$$\langle \mathbf{c}, \bar{\mathbf{x}} \rangle \leqslant \left(1 + \frac{C}{2}\alpha_J\right) OPT^E, \tag{2.22}$$

and for each $i \in I, k \in K$*, and*

$$\sum_{m:(i,k)\in m} \bar{x}_m - S_{ik} \leqslant \frac{\bar{C}}{\sqrt{J\alpha_J}}, \tag{2.23}$$

for positive constants C*,* $\bar{C}$ *independent of* J*.*

Again, note that, as we show at the end of the proof of Theorem 2.2, the constant $\bar{C}$ always exists (and the constant C is the one from Proposition 2.2). Moreover, note that in Theorem 2.2, in order to balance the convergence rate of both (2.22) and (2.23), we can choose the step-size to be $\alpha_J = J^{-1/3}$, obtaining a convergence rate of $J^{-1/3}$. Because of this reason, in what follows, we will use $\alpha_J = J^{-1/3}$ as our step-size.

It is worth noting that, according to Theorem 2.2, the (scaled) average primal solution $\bar{\mathbf{x}}$ computed by Algorithm 3 (at iteration J) is close to being optimal, i.e., it is an ϵ-approximation. However, given (2.23), Theorem 2.2 does not guarantee that $\bar{\mathbf{x}}$ satisfies the inventory constraint of problem (2.10). Fortunately, one can construct a feasible solution for (2.10) as follows.

Let $\bar{\mathbf{x}}$ denote the output of Algorithm 3, i.e.

$$\bar{\mathbf{x}} = \frac{1}{J}\sum_{j=1}^{J}\sum_{q\in Q} p^a(q)\mathbf{x}[q]^{(j)} = \left(\frac{1}{J}\sum_{j=1}^{J}\sum_{q\in Q} p^a(q)x_m^{(j)}\mathbb{1}\{m \sim q\}\right)_{m\in M},$$

with each element denoted as $\bar{x}_m$, for $m \in M$. Then, $\bar{\mathbf{x}}$ satisfies constraint (2.11) of

problem (2.10). Indeed, for $q' \in Q$

$$\begin{aligned}
\sum_{m:m\sim q'} \bar{x}_m &= \sum_{m:m\sim q'} \left(\frac{1}{J} \sum_{j=1}^{J} \sum_{q\in Q} p^a(q) x_m^{(j)} \mathbb{1}\{m \sim q\} \right) \\
&= \sum_{m\in M} \left(\frac{1}{J} \sum_{j=1}^{J} \sum_{q\in Q} p^a(q) x_m^{(j)} \mathbb{1}\{m \sim q\} \mathbb{1}\{m \sim q'\} \right) \\
&= \frac{1}{J} \sum_{j=1}^{J} \sum_{q\in Q} p^a(q) \left(\sum_{m\in M} x_m^{(j)} \mathbb{1}\{m \sim q\} \right) \mathbb{1}\{q = q'\} \\
&\overset{(a)}{=} \frac{1}{J} \sum_{j=1}^{J} \sum_{q\in Q} p^a(q) \mathbb{1}\{q = q'\} \\
&= p^a(q') \qquad (2.24)
\end{aligned}$$

where (a) holds because $\mathbf{x}[q]^{(j)}$ is a feasible solution to $L_q(\boldsymbol{\lambda}^{(j)})$ at iteration j of Algorithm 3 and so, by definition, $\sum_{m:m\sim q} x_m^{(j)} = 1$. Given this, we now show how to construct a solution for problem (2.10) that satisfies the inventory constraint. Let $\bar{\mathbf{x}} = \{\bar{x}_m\}_{m\in M}$ be defined as before and let

$$\tau := \max \left\{ 1, \max_{i\in I, k\in K, S_{ik}>0} \frac{\sum_{m:(i,k)\in m} \bar{x}_m}{S_{ik}} \right\} \qquad (2.25)$$

be its largest violation of the inventory constraint. For an order type $q \in Q$ and denoting by $m'(q)$ the discard method for q, define

$$y_m = \begin{cases} \bar{x}_m \cdot \tau^{-1} & \text{if } m \neq m'(q) \\ p^a(q) \cdot \left(1 + \tau^{-1} \frac{\bar{x}_{m'(q)} - \sum_{m:m\sim q} \bar{x}_m}{\sum_{m:m\sim q} \bar{x}_m} \right) & \text{if } m = m'(q). \end{cases} \qquad (2.26)$$

Then, $\{y_m\}_{m\in M}$ is feasible for problem (2.10). Indeed, for a given $q \in Q$, we have that

$$\sum_{m:m\sim q} y_m = p^a(q).$$

Moreover,

$$\sum_{m:(i,k)\in m} y_m \overset{(a)}{=} \sum_{m:(i,k)\in m} \bar{x}_m \tau^{-1} \overset{(b)}{\leqslant} S_{ik},$$

where (a) holds because the sum does not involve the discard method $m'(q)$ (i.e., the inventory constraint is automatically satisfied for the dummy facility); and (b) follows from the definition of τ in (2.25).

Finally, we show that $\mathbf{y} = \{y_m\}_{m \in M}$ only results in a small sacrifice of the competitive ratio of our fulfillment strategy. Our main result of this section is formally stated next.

Theorem 2.3. *Suppose we use* $\mathbf{y} = \{y_m\}_{m \in M}$ *as defined in* (2.26) *to find a feasible solution to problem* (2.10) *in the offline process of Algorithm 1. Then, for any online multi-item fulfillment problem, when Algorithm 1 uses Algorithm 2 as a subroutine in Step 2 of the online process, we obtain a fulfillment strategy with a competitive ratio of at most*

$$\left(\frac{sJ^{1/3} + \kappa\bar{C}}{sJ^{1/3} + \bar{C}}\right)\left(1 + \frac{C}{2J^{1/3}}\right)\left(1 + \frac{(\kappa - 1)|q_{\max}|}{\sqrt{s+3}}\right). \tag{2.27}$$

where J *is the number of iterations of the supergradient method defined by Algorithm 3, and* $\bar{C}$ *and* C *are some positive constants from Theorem 2.2 that are independent of* J*. Alternatively, the competitive ratio is at most*

$$(1 + \epsilon)\left(1 + \frac{(\kappa - 1)|q_{\max}|}{\sqrt{s+3}}\right), \tag{2.28}$$

with $\epsilon \leqslant \widetilde{C} J^{-1/3}$*, where* $\widetilde{C}$ *is some positive constant that is independent of* J*.*

Note that, according to (2.27), as J (the number of iterations in Algorithm 3) and s (the amount of minimum inventory) go to infinity, our fulfillment strategy with the approximate solution is asymptotically optimal. It is also worth noting that Theorem 2.3 implies a trade-off between computation and performance of our policy: by increasing J we need more offline computations, but we obtain a better solution to problem (2.10) and a better competitive ratio of our fulfillment strategy. Therefore,

our analysis shows that there is a trade-off between the computations required to obtain the online policy and its performance.

We finish this section by studying a problem raised by Jasin and Sinha (2015) using our supergradient method. Specifically, in the additive cost framework of Jasin and Sinha (2015) (described in Section 2.10), our method, combined with a novel randomized rounding, leads to a policy that is asymptotically $O(\log |q_{\max}|)$-competitive and runs polynomially in $|q_{\max}|$, $|K|$ and $|Q|$.

Proposition 2.3. *Consider the additive cost multi-item fulfillment framework. Then, for this framework, there exists an algorithm that is asymptotically $O(\log |q_{\max}|)$-competitive, as $s \to \infty$, where s is the minimum level of inventory for any item.*

The details and the proof of the above result are provided in Section 2.10.2 (Theorem 2.5). In particular, in Section 2.10.2, we provide the explicit algorithm that is asymptotically $O(\log |q_{\max}|)$-competitive, as $s \to \infty$. We obtain this result by first considering the Lagrangian relaxation of the corresponding offline LP which allows us to decompose this LP into problems that involve only a single order. Then, we reformulate the decomposed LPs using the item-facility framework of Jasin and Sinha (2015) and develop a novel sequential randomized rounding scheme to approximately solve them.

2.6 Connection with Network Revenue Management

In this section, we discuss the connections and the implications of our analysis in the network revenue management (NRM) literature. Network revenue management is a class of online resource allocation problems in which items (or resources) with limited inventories (or capacities) are sold to satisfy the orders of a stream of customers arriving sequentially over time. Specifically, at each time period, a customer arrives with an order for one or multiple types of items. If we fulfill the order request, we

collect some revenue and consume the inventories corresponding to the items in the order. The goal is to find a policy to decide which orders to accept in order to maximize the total expected revenue over the entire selling horizon.

We consider the independent arrival model, with $\mathcal{D}^{qt}$ denoting the number of arrivals of order type q at time t, i.e., $\mathcal{D}^{qt}$ is a Bernoulli random variable with success probability $p_t(q)$. Moreover, we let $x_q^t \in \{0,1\}$ denote the binary variable representing the offline decision of fulfilling order q at time t, i.e., $x_q^t = 1$ if the platform fulfills order q at time t and $x_q^t = 0$, otherwise. We also denote by r_q the revenue/reward for fulfilling order type q. Then, the offline problem associated with a NRM problem can be formulated as follows

$$\max_{\mathbf{x}} \sum_{t\in[T]} \sum_{q\in Q} r_q x_q^t \tag{2.29}$$

$$\text{s.t. } x_q^t \leqslant \mathcal{D}^{qt}, \ \forall q \in Q, t \in [T], \tag{2.30}$$

$$\sum_{t\in[T]} \sum_{q:i\in q} x_q^t \leqslant S_i, \ \forall i \in I, \tag{2.31}$$

$$x_q^t \in \{0,1\}, \ \forall q \in Q, t \in [T]. \tag{2.32}$$

In this formulation, the objective is the platform's profit, which is the sum of collected rewards from all periods; the set of constraints (2.30) ensures that we fulfill order q only when it arrives, allowing the option of not fulfilling the order; and, finally, the set of constraints (2.31) ensures that, over the entire time horizon $[T]$, we do not fulfill orders using more than S_i units of item $i \in I$. The LP relaxation associated

with (2.29) is

$$\max \sum_{t\in[T]} \sum_{q\in Q} r_q x_q^t \tag{2.33}$$

$$\text{s.t. } x_q^t \leqslant p_t(q),\ \forall q \in Q, t \in [T], \tag{2.34}$$

$$\sum_{t\in[T]} \sum_{q:i\in q} x_q^t \leqslant S_i,\ \forall i \in I, \tag{2.35}$$

$$x_q^t \in [0,1],\ \forall q \in Q, t \in [T], \tag{2.36}$$

where x_q^t now can be interpreted as the probability of fulfilling order type q at time t. Note that, the standard NRM formulation may be viewed as a special case of our fulfillment model that maximizes the revenue when there is a single facility and each order can only be fully fulfilled or lost.

It is also worth mentioning that an important characteristic of NRM problems is the trade-off between accepting an order to generate some immediate revenue and saving inventories for potentially more profitable orders that can arrive in the future. Because of this trade-off, in these types of problems, the focus is on accept/reject decisions based on the available inventories. This closely relates to our method-acceptance problem. More specifically, a NRM problem has a similar decision structure to our method-acceptance problem, except that it does not assume the expected demand to not exceed the available inventory. Given this connection between NRM problems and our method-acceptance problem, we next show (in the proof) that Algorithm 4, using a γ-conservative method-acceptance strategy combined with randomization (using the solution of the LP relaxation), directly implies a competitive ratio of at least γ for NRM problems.

Proposition 2.4. *For any network revenue management problem, when Algorithm 4 uses Algorithm 2 (in the special case with one facility) as a subroutine in Step 2 of*

Algorithm 4 Algorithm for NRM

Offline Process:

- Solve the expected LP in (2.33) to obtain y_q^t for all $q \in Q$ and $t \in [T]$.

Online Process:

- **For** $t = 1, \ldots, T$:

  Let q be the order that arrives at time t.

  1. Draw $Y_q^t \sim \text{Bernoulli}(y_q^t / p_t(q))$.
  2. If $Y_q^t = 1$, then decide whether to accept or reject order q using a γ-conservative method-acceptance strategy. If $Y_q^t = 0$, reject order q.

---

the online process, we obtain a policy with a competitive ratio of at least

$$1 - \frac{|q_{\max}|}{\sqrt{s+3}},$$

with $|q_{\max}|$ *denoting the size of the largest possible order and* $s = \min_{i \in I}\{S_i\} \geqslant |q_{\max}|^2 - 3$.

We provide the proof of the above statement in Section 2.13. Note that Proposition 2.4 provides an algorithm for NRM problems whose competitive ratio depends on $|q_{\max}|$ and s. It is worth highlighting that the state-of-the-art guarantees (in terms of competitive ratio) for NRM problems are the ones by Ma et al. (2020) in which the authors prove $1/(1 + |q_{\max}|)$-competitiveness; and Baek and Ma (2022), where the authors generalize and improve the $1/(1 + |q_{\max}|)$ result by exploiting particular network structures (while also recovering the same guarantee as a special case). To the best of our knowledge, Proposition 2.4 identifies the first algorithm with a competitive ratio that depends on s and in which the ratio converges to 1 as s goes to infinity (when $|q_{\max}|$ is fixed). Remember that s is the minimum level of inventory. The condition of $s \geqslant |q_{\max}|^2 - 3$ is to ensure that the competitive ratio is not vacuous.

We should mention that this is a drawback of our analysis as the above bounds in the literature work for any s. Extending the analysis so that the competitive ratio becomes $1/(1+q_{\max})$ for small s (similar to Ma et al. (2020) and Baek and Ma (2022)) and becomes asymptotically optimal as $s \to \infty$ is an interesting future direction to explore. When s is much larger than $|q_{\max}|$, Proposition 2.4 provides a significantly higher competitive ratio compared to the current results in the literature. Moreover, Proposition 2.4 also implies that Algorithm 4 is asymptotically optimal in the classical "fluid-scaling" regime, where both time and inventory are scaled to infinity.

2.7 Numerical Simulation

In this section, we perform numerical simulations to evaluate our magician-based policy designed for the multi-item fulfillment model described in Section 4.2. Our primary objective is to compare our policy with the state-of-the-art correlated rounding policy of Ma (2023). This comparison is directed towards understanding the performance implications of employing a distinct offline problem formulation (a method-based formulation as opposed to an item-facility-based formulation), as described in (2.1). To ensure consistency and relevance in our comparisons, we have aligned our simulation environment as closely as possible with the framework utilized by Ma (2023). Our experiments are conducted using the Python programming language and the Gurobipy package. Details of our simulation environment, including data and code, are provided in Section 2.11. Briefly, the simulation environment that we consider is similar to a setup studied in Ma (2023), with an e-commerce network with the 10 largest U.S. cities spread geographically across the country and 5 centrally located facility centers. We assume that the online retailer has 20 different item types and let the time horizon be $T = 1000$. Order arrivals are i.i.d. and orders vary in size from 1 to a maximum size $n_{\max}$ of either 2 or 5, with each order size having $n_0 = 5$ distinct possible combination of items (orders of size zero, i.e., no order, are also considered).

For each order size, a unique combination of items is randomly selected from I. Then, for each combination of items, $|J|$ orders are generated, one for each city, leading to a total of $|J|(1 + n_{\max} n_0)$ different order types.

Our parameterized policy. Remember that there are two main building blocks to our policy: (1) the adoption of a probabilistic method-acceptance strategy; and (2) a different formulation of the offline problem (with respect to Ma (2023)), which we call method-based formulation. To isolate the impact of each of these building blocks, we introduce a parametrized family of policies. Specifically, we define $\nu \in [0, 1]$ to be a constant that influences the conservativeness of each magician's decision-making process. This modification alters the description of Algorithm 2 by changing its inputs from $\gamma_{ik} = 1 - 1/\sqrt{S_{ik} + 3}$ to $\widetilde{\gamma}_{ik} = 1 - \nu/\sqrt{S_{ik} + 3}$. Recall that γ_{ik} represents the ex-ante probability (prior to any order arrivals) of magician (i, k) making their inventory available for fulfillment. We change this probability to $\widetilde{\gamma}_{ik}$, thus allowing ν to effectively model the magicians' decision-making conservativeness. Indeed, the case of $\nu = 0$ corresponds to the "always accept" policy (i.e., the policy that always accepts a method as long as there is sufficient inventory), characterized by minimal conservativeness, while $\nu = 1$ fully captures our magician-based approach. For values of $\nu \in (0, 1)$, this parameterized policy exhibits varying degrees of conservativeness, bridging the two extremes. It is important to acknowledge, especially in the context of the findings from Alaei (2014), that $\nu \in (0, 1)$ does not guarantee the availability of inventory until the end of the time horizon. Thus, this modification introduces a heuristic element to our strategy, necessitating a flexible approach that includes verifying the availability of inventory before fulfilling orders. By incorporating this parameterization, our aim is to assess the robustness and practical effectiveness of our fulfillment algorithm. In Table 2.2, we present the average cost of our policy when the maximum number of items in an order are $n_{\max} = 2$ and $n_{\max} = 5$. These

average costs are taken over 30 instances, where each instance differs by order types, demand rates, and initial inventories (again, we refer to Section 2.11 for details), and over 100 draws of sequences of order arrivals for each of instance. Finally, note that in our numerical study, we use the variation of Algorithm 2, described in Remark 2.1 so that we do not incur any additional costs for $(i, k) \in m$ when some other magician $(i', k') \in m$ does not make their inventory available.

To highlight the impact of adopting a probabilistic method-acceptance strategy, we observe that in Table 2.2, by decreasing the conservativeness of every magician, i.e., as ν decreases, our policy performs better. In particular, a policy that always accepts methods as long as there is available inventory, i.e., when $\nu = 0$, has the best performance. The reason for this observation is twofold. First, in our simulation, the inventory level S_{ik} for all $i \in I$ and $k \in K$ scales linearly in the time horizon T and proportionally to the average demand. Therefore, the probability of running out of inventory is small. This makes a more conservative magician-based method acceptance policy less effective as it is designed to work for scarce resources and to be oblivious to the time horizon (see Section 2.9 for more discussion of the original magician problem). In particular, for this i.i.d. simulation setting, holding inventory for the future offers no real value, making the magician-based method-acceptance policy overly conservative. Second, the magician-based acceptance method is designed to guarantee a competitive ratio for the worst case. This means that even in settings where the lost sale is much larger than the cost of fulfilling an order and the orders are arriving in an arbitrary way, the magician-based acceptance method guarantees that we fulfill each order with a large enough probability. However, in practice and in our numerical studies, the cost of not fulfilling an order is comparable to the cost of fulfilling it, making smaller ν a better choice. For instance, in Jasin and Sinha (2015) and Zhao et al. (2020), as well as in the more recent paper of Ma (2023), the authors assume that each unfulfilled item incurs a lost-sale penalty cost equal

to twice the maximum single-item cost. Finally, we observe that, although in the i.i.d. order arrival setting the "always accept" policy outperforms the magician-based policy, there are cases in which adopting a conservative approach is beneficial. For example, when orders of larger size arrive later during the time horizon, our policy can outperform the "always accept" policy. To verify this, we simulate arrival sequences in which in the first $\lfloor T/2 \rfloor$ periods only orders of size 1 can arrive, and in the remaining periods only orders of size larger than 1 can arrive. Under this setting, when $\nu = 0.01$ and when we use a γ_{ik}-conservative magician policy for a pair (i, k) that is requested throughout the entire time horizon, the expected cost of our magician-based policy is 10418.54, while the cost of the "always accept" policy is 10418.98, providing a reduction in cost of about 0.004%. The reason for this improvement is that magician (i, k) is now used for inventory that is valuable (because it is high in demand), i.e., (i, k) is requested in multiple orders throughout the entire time horizon.

On the other hand, to highlight the impact of adopting a method-based offline formulation, we observe that, for small values of ν (i.e., when the magicians are less conservative), our policy outperforms that of Ma (2023) in terms of the expected cost. Let us explain this difference and also compare the two policies from other perspectives. The policy of Ma (2023) uses an item-facility offline formulation and achieves the *optimal* competitive ratio with respect to that offline formulation. Instead, we use a method-based offline formulation and show how to achieve a (not necessarily optimal) competitive ratio with respect to that offline formulation. The gain of our policy stems from this alternative formulation. However, as we have discussed earlier in the chapter, this comes at the cost of requiring more computations to find the optimal offline solution (that is needed for our policy). To see this, as shown in Table 2.3, for $n_{\max} = 5$, the running time to find the optimal offline solution of our LP formulation is .51 seconds while the running time to find the optimal offline solution to the LP described in Ma (2023) is .17 seconds which is smaller. For larger values of $n_{\max}$,

solving our LP formulation takes even longer time, necessitating using other methods such as the supergradient method we developed in Section 2.5. Moreover, note that in Table 2.3, for convenience, the number of variables for our method-based formulation are counted by only considering methods (as defined in Section 4.2) that could be used for a given order type.

Table 2.2: Average costs with different ν and $n_{\max}$ values.

$n_{\max}$	**Policy**	**Cost**
$n_{\max} = 2$	$\nu = 0$	8197.95
	$\nu = 0.01$	8201.49
	$\nu = 0.1$	8250.88
	$\nu = 1$	9629.14
	Ma (2023)	8316.76
$n_{\max} = 5$	$\nu = 0$	13001.67
	$\nu = 0.01$	13014.65
	$\nu = 0.1$	13180.19
	$\nu = 1$	16390.24
	Ma (2023)	14239.88

Table 2.3: Comparison of our LP and Ma (2023)'s LP for different $n_{\max}$ values.

$n_{\max}$	**LP**	**Runtime**	**Opt. Value**	**#Variables**	**#Constraints**
$n_{\max} = 2$	Method-based	0.003	7522.58	1425	210
	Ma (2023)	0.017	7522.58	13860	1150
$n_{\max} = 5$	Method-based	0.528	11454.99	159488	360
	Ma (2023)	0.177	11310.84	32760	5350

2.8 Conclusion

In this chapter, we study the multi-item e-commerce fulfillment problem from the literature. We propose a fulfillment policy for the problem that achieves $1 + (\kappa - 1)|q_{\max}|/\sqrt{s+3}$ competitive ratio, the first known non-asymptotic competitive ratio that depends only on the order size, inventories and κ, the maximum ratio between the cost of not fulfilling an order and the cost of fulfilling it with any other method. In addition, our competitive ratio is independent of the number of item and order types, and our analysis derives, to the best of our knowledge, the first non-asymptotic approximation guarantee that is also asymptotically optimal under the so-called "fluid scaling" regime.

Our approach to solving the multi-item fulfillment problem combines multiple tools from different works of literature. First, we propose an algorithm that consists of two key components: a probabilistic fulfillment step and an accept/reject step (which we call method-acceptance problem). This accept/reject step helps us control the on-hand inventory and connect key ideas in the fulfillment and prophet inequality literature. Second, we apply the strategies from Alaei (2014) to show that one can design a $(1 - |q_{\max}|/\sqrt{s+3})$-conservative method-acceptance strategy. Then, we combine this strategy with probabilistic fulfillment to show that our algorithm achieves the desired competitive ratio for the multi-item fulfillment problem. It is important to note that our procedure does not require Alaei (2014)'s result. For example, we can alternatively use k-unit OCRS as a subroutine, and our result can be improved using the tight bound proved by Jiang et al. (2021). In general, we show that as long as you have a γ-conservative method-acceptance strategy, one can achieve a competitive ratio of $1 + (\kappa - 1)(1 - \gamma)$ (and when $\gamma = 1 - |q_{\max}|/\sqrt{s+3}$ this reduces to our result). For simplicity of exposition, we adopt the closed-form guarantee from Alaei (2014). In Section 2.6, we also note that a special case of our algorithm provides

new asymptotically optimal bounds for network revenue management problems (see, e.g., Ma et al., 2020; Baek and Ma, 2022) where the focus is on accept/reject decisions about the available resources (see Section 2.6 for further details).

Finally, in order to perform the probabilistic fulfillment step, we develop a super-gradient method to approximately solve the LP relaxation of the multi-item fulfillment problem, which is computationally effective when $(|K|+1)^{|q|}$ is small for each order q. A similar method, combined with a randomized rounding algorithm, also leads to a policy that is asymptotically $O(\log|q_{\max}|)$-competitive (as $s \to \infty$) for the additive cost multi-item fulfillment framework of Jasin and Sinha (2015), answering a question raised by the authors. We note that a similar $O(\log|q_{\max}|)$ result was recently derived by Ma (2023) using different techniques than ours.

We conclude by noting several interesting future directions to explore. For example, in our model, we have considered the case in which we do not have replenishment throughout the fulfillment process. An interesting future direction could be incorporating and modeling replenishment decisions in our fulfillment policy. As another example, we assumed that the platform only knows the distribution of the arriving orders and has no additional information about their arrival. In many online settings, however, the platform may have partial information about the arrival of orders. Modeling this additional information in our framework and incorporating it in the fulfillment strategy is an exciting future direction to explore.

2.9 A γ_{ik}-Conservative Strategy for the Magician (i,k) Problem

In this section, we present a self-contained overview of a γ_{ik}-conservative strategy for the magician (i,k) problem presented in Definition 2.4. We first describe the original magician problem from Alaei (2014) and then show the details of one such strategy for Definition 2.4, inspired by Alaei (2014).

Magician problem. Alaei (2014) describes the magician problem as follows:

A magician is presented with a sequence of boxes one by one, in an online fashion. There is a prize hidden in one of the boxes. The magician has k magic wands that can be used to open the boxes. If a wand is used on box i, it opens, but with a probability of at most x_i, which is written on the box, the wand breaks. The magician wants to maximize the probability of collecting the prize, but the sequence of boxes, the written probabilities, and the box in which the prize is hidden are arranged by a villain, and the magician has no prior information about them. However, it is guaranteed that $\sum_i x_i \leqslant k$ and that the villain has to prepare the sequence of boxes in advance.

Alaei (2014) presents a strategy, called γ-conservative strategy, that guarantees that the probability of finding the prize is at least $\gamma = 1 - \frac{1}{\sqrt{k+3}}$. This strategy works for any number of boxes n and any number of wands k (that can potentially be much smaller than n). Moreover, this strategy does not require the knowledge of n and works as long as the inequality $\sum_i x_i \leqslant k$ holds.

We next explain the high-level connection between the above problem and our fulfillment problem and then provide the details. The connection is not so much in the description of the two problems but rather in the solution used for solving the two. In particular, in the magician problem, because the magician does not know the whereabouts of the prize, the solution (i.e., the magician's strategy) must be conservative to guarantee that the probability of opening each box is large (so that the probability of collecting the reward is large), knowing that the magician has a limited number of wands. Now, in our fulfillment problem, the solution must be conservative so that we guarantee to fulfill any arriving order with a large probability, knowing that we have limited inventory. We next formalize this connection and introduce a conservative strategy for any item-facility pair (i, k).

γ_{ik}-conservative strategy. For this strategy, we assume that we can solve (at least approximately) the expected LP in (2.10) and obtain the corresponding $\mathbf{y} = \{y_m^t\}_{m \in M, t \in [T]}$ through (2.14). See Section 2.5 for the details of how to solve (2.10) approximately.

For each magician (i,k) in Definition 2.4, let $\mu_{ik}^t = \sum_{m:(i,k)\in m} y_m^t$ and let W_{ik}^t denote the number of consumed inventory (i,k) (i.e. of item i from facility k) prior to period t. Then, magician (i,k) makes a decision about making their inventory available for fulfillment by comparing W_{ik}^t and a threshold θ_{ik}^t in the following way:

$$\begin{cases} \text{make } (i,k) \text{ available} & \text{if } W_{ik}^t < \theta_{ik}^t; \\ \text{make } (i,k) \text{ available with some probability (to be defined)} & \text{if } W_{ik}^t = \theta_{ik}^t; \\ \text{do not make } (i,k) \text{ available} & \text{if } W_{ik}^t > \theta_{ik}^t. \end{cases}$$

Magician (i,k) chooses θ_{ik}^t to be the smallest threshold l such that $\mathbb{P}(W_{ik}^t \leqslant l) \geqslant \gamma_{ik}$, where the probability $\mathbb{P}(W_{ik}^t \leqslant l)$ can be computed before period t for any l (because of the definition of W_{ik}^t). Formally, let $F_{W_{ik}^t}(l) = \mathbb{P}(W_{ik}^t \leqslant l)$ be the CDF of the random variable W_{ik}^t and let A_{ik}^t be the indicator variable which is 1 if and only if magician (i,k) makes their inventory available at time t. Then, we can rewrite the probability with which magician (i,k) should make their inventory available, conditional on W_{ik}^t, as follows:

$$\mathbb{P}(A_{ik}^t = 1 \mid W_{ik}^t) = \begin{cases} 1 & \text{if } W_{ik}^t < \theta_{ik}^t; \\ (\gamma_{ik} - F_{W_{ik}^t}(\theta_{ik}^t - 1))/(F_{W_{ik}^t}(\theta_{ik}^t) - F_{W_{ik}^t}(\theta_{ik}^t - 1)) & \text{if } W_{ik}^t = \theta_{ik}^t; \\ 0 & \text{if } W_{ik}^t > \theta_{ik}^t; \end{cases}$$

$$\theta_{ik}^t = \min\left\{ l : F_{W_{ik}^t}(l) \geqslant \gamma_{ik} \right\}.$$

Note that the threshold θ_{ik}^t can be computed before period t (because of the definition of $F_{W_{ik}^t}(\cdot)$). Now, define $a_{ik}^{tl} := \mathbb{P}(A_{ik}^t = 1 \mid W_{ik}^t = l)$. Then, it is easy to see that the CDF of $W_{ik}^{(t+1)}$ can be computed from the CDF of W_{ik}^t and μ_{ik}^t as follows

(assuming μ_{ik}^t is the exact probability of consuming the inventory at time t):

$$F_{W_{ik}^{(t+1)}}(l) = \begin{cases} a_{ik}^{tl}\mu_{ik}^t F_{W_{ik}^t}(l-1) + (1 - a_{ik}^{tl}\mu_{ik}^t)F_{W_{ik}^t}(l) & \text{if } t \geqslant 1, l \geqslant 0; \\ 1 & \text{if } t = 0, l \geqslant 0. \\ 0 & \text{otherwise.} \end{cases}$$

Moreover, if each μ_{ik}^t is just an upper bound on the probability of consuming the inventory at time t, then the above definition of $F_{W_{ik}^t}$ stochastically dominates the actual CDF of W_{ik}^t (i.e. $F_{W_{ik}^t}(l) \leqslant \mathbb{P}(W_{ik}^t \leqslant l)$ for all l), and magician (i,k) makes their inventory available with probability at least γ_{ik}. This can be proven formally by induction. Indeed, the base case $t = 1$ trivially holds (because both quantities are 1). Suppose now that the inequality holds for $t \geqslant 1$. We prove that it holds for $t+1$ as follows:

$$\begin{aligned} \mathbb{P}(W_{ik}^{(t+1)} \leqslant l) &\geqslant \mathbb{P}(W_{ik}^t \leqslant l-1) + \mathbb{P}(W_{ik}^t = l)(1 - a_{ik}^{tl}\mu_{ik}^t) \\ &= \mathbb{P}(W_{ik}^t \leqslant l-1) + [\mathbb{P}(W_{ik}^t \leqslant l) - \mathbb{P}(W_{ik}^t \leqslant l-1)](1 - a_{ik}^{tl}\mu_{ik}^t) \\ &= \mathbb{P}(W_{ik}^t \leqslant l-1)a_{ik}^{tl}\mu_{ik}^t + \mathbb{P}(W_{ik}^t \leqslant l)(1 - a_{ik}^{tl}\mu_{ik}^t) \\ &\geqslant F_{W_{ik}^t}(l-1)a_{ik}^{tl}\mu_{ik}^t + F_{W_{ik}^t}(l)(1 - a_{ik}^{tl}\mu_{ik}^t) = F_{W_{ik}^{(t+1)}}(l). \end{aligned}$$

Finally, note that the randomization probability (in the case in which $W_{ik}^t = \theta_{ik}^t$) is defined in a way so that the ex-ante probability of opening each box is at least γ_{ik}. Indeed,

$$\begin{aligned} \mathbb{P}(A_{ik}^t = 1) &= \sum_{l=0}^{t-1} a_{ik}^{tl}\mathbb{P}(W_{ik}^t = l) = \sum_{l=0}^{\theta_{ik}^t} a_{ik}^{tl}\mathbb{P}(W_{ik}^t = l) \\ &= \mathbb{P}(W_{ik}^t < \theta_{ik}^t) + a_{ik}^{t\theta_{ik}^t}\mathbb{P}(W_{ik}^t = \theta_{ik}^t) \\ &= \mathbb{P}(W_{ik}^t \leqslant \theta_{ik}^t - 1) + a_{ik}^{t\theta_{ik}^t}[\mathbb{P}(W_{ik}^t \leqslant \theta_{ik}^t) - \mathbb{P}(W_{ik}^t \leqslant \theta_{ik}^t - 1)] \\ &\geqslant (1 - a_{ik}^{t\theta_{ik}^t})F_{W_{ik}^t}(\theta_{ik}^t - 1) + a_{ik}^{t\theta_{ik}^t}F_{W_{ik}^t}(\theta_{ik}^t) \\ &= F_{W_{ik}^t}(\theta_{ik}^t - 1) + a_{ik}^{t\theta_{ik}^t}[F_{W_{ik}^t}(\theta_{ik}^t) - F_{W_{ik}^t}(\theta_{ik}^t - 1)] = \gamma_{ik}. \end{aligned}$$

Under this strategy, (Alaei, 2014, Theorem 4) guarantees that, whenever $\gamma_{ik} \leqslant 1 - 1/\sqrt{S_{ik}+3}$, magician (i,k) never requires more than S_{ik} inventory and they are guaranteed an ex-ante probability (before the fulfillment process starts) of at least γ_{ik} of making the inventory available at any time period t.

2.10 Item-Facility-Based Model of Jasin and Sinha (2015)

In Jasin and Sinha (2015), the authors assume that the cost for each method m can be decomposed into fixed and variable costs of shipping items from facilities. We can adopt the same assumption in our method-based framework as follows. For a given method m, define $K_m := \{k \in K : (i,k) \in m\}$ to be the set of facilities used for fulfillment by method m. Then, the cost of using method m for its corresponding order type can be written as:

$$c_m = \sum_{(i,k)\in m} v_{ik}^q + \sum_{k\in K_m} b_k^q, \tag{2.37}$$

where v_{ik}^q and b_k^q denote the unit variable cost and the fixed cost associated with shipping items from facility k to the location where order type q is placed, respectively.

Therefore, changing the variables x_m^t to x_{ik}^{qt} (with x_{ik}^{qt} denoting the binary variable representing the decision of shipping item i from facility k to satisfy order type q at time t), and defining $u_{ik}^q := \sum_{m:m\sim q} \mathbb{1}\{(i,k) \in m\}$ and $w_k^q := \sum_{m:m\sim q} \mathbb{1}\{k \in K_m\}$ (used in objective 2.38 to avoid over-counting of fixed and variable costs whenever $x_{ik}^{qt} = 1$ and pair (i,k) is in more than one method for order type q), our offline problem can

be reformulated as follows:

$$\min_{\mathbf{x}} \sum_{t\in[T]} \sum_{q\in Q} \sum_{m:m\sim q} \left(\sum_{(i,k)\in m} \frac{1}{u_{ik}^q} v_{ik}^q x_{ik}^{qt} + \sum_{k\in K_m} \frac{1}{w_k^q} b_k^q \max_{i\in q} x_{ik}^{qt} \right) \tag{2.38}$$

$$\text{s.t.} \sum_{k\in K} x_{ik}^{qt} = \mathcal{D}^{qt},\ \forall i \in q, q \in Q, t \in [T],$$

$$\sum_{t\in[T]} \sum_{q\ni i} x_{ik}^{qt} \leqslant S_{ik},\ \forall i \in I, k \in K,$$

$$x_{ik}^{qt} \in \{0,1\},\ \forall i \in I, k \in K, q \in Q, t \in [T].$$

Now, note that there is a close connection between x_m^t and the decision variables $\{x_{ik}^{qt}\}_{i\in I,k\in K}$ in the item-facility based model of Jasin and Sinha (2015): for a given method m for order type q, we have that whenever $x_m^t = 1$, then $x_{ik}^{qt} = 1$ for $(i,k) \in m$ and $x_{ik}^{qt} = 0$, otherwise. We further discuss this connection in the next section.

2.10.1 Sequential Randomized Rounding Algorithm

In this section, we show how our supergradient method leads to a policy that is asymptotically $O(\log|q_{\max}|)$-competitive (as $s \to \infty$) for the general multi-item fulfillment framework of Jasin and Sinha (2015), which specifically write: "... this still leaves open a gap between our competitive ratio of $\mathbb{E}[B(|\mathcal{Q}|)]$ and the inapproximability threshold of $\Omega(\log|Q|)$. The standard techniques used to approximate the set cover problem do not directly extend to our problem; the main difficulty lies in the capacity constraints. Reducing this gap remains an open question."

In Section 2.5, we showed how to obtain a computationally viable method for approximately solving (2.10) through a supergradient method for its dual problem (as (2.10) decomposes into many smaller LPs under a Lagrangian relaxation). For the reader's convenience, we rewrite here the Lagrangian relaxation that we want to solve and its decomposition into smaller LPs. For $\boldsymbol{\lambda} = \{\lambda_{ik}\}_{i\in I,k\in K} \in \mathbb{R}_+^{|I||K|}$, we define the Lagrangian function $L(\boldsymbol{\lambda}, \mathbf{x}) := \sum_{q\in Q}\langle \mathbf{c}, \mathbf{x}[q]\rangle + \sum_{i\in I,k\in K} \lambda_{ik}(\sum_{m:(i,k)\in m} x_m - S_{ik})$,

and the Lagrangian relaxation as follows:

$$L(\boldsymbol{\lambda}) \coloneqq \min_{\mathbf{x}} L(\boldsymbol{\lambda}, \mathbf{x}) \tag{2.39}$$

$$\text{s.t.} \quad \sum_{m:m\sim q} x_m = p^a(q), \ \forall q \in Q,$$

$$x_m \in [0, T], \ \forall m \in M.$$

As noted in (2.18), the above Lagrangian relaxation can be rewritten as:

$$L(\boldsymbol{\lambda}) = \min_{\mathbf{x}} \sum_{q\in Q} p^a(q) \left(\langle \mathbf{c}, \mathbf{x}[q] \rangle + \sum_{i\in q, k\in K} \lambda_{ik} \sum_{m:(i,k)\in m, m\sim q} x_m \right) - \sum_{i\in I, k\in K} \lambda_{ik} S_{ik}$$

$$\text{s.t.} \quad \sum_{m:m\sim q} x_m = 1, \ \forall q \in Q,$$

$$x_m \in [0, 1], \ \forall m \in M.$$

Moreover, one can check that

$$L(\boldsymbol{\lambda}) = \sum_{q\in Q} p^a(q) L_q(\boldsymbol{\lambda}) - \sum_{i\in I, k\in K} \lambda_{ik} S_{ik},$$

where, for each $q \in Q$,

$$L_q(\boldsymbol{\lambda}) \coloneqq \min_{\mathbf{x}[q]} \langle \mathbf{c}, \mathbf{x}[q] \rangle + \sum_{i\in q, k\in K} \lambda_{ik} \left(\sum_{m:(i,k)\in m, m\sim q} x_m \right) \tag{2.40}$$

$$\text{s.t.} \quad \sum_{m:m\sim q} x_m = 1,$$

$$x_m \in [0, 1], \ \forall m \in M \text{ such that } m \sim q.$$

This shows that (2.39) decomposes across Q into smaller LPs of the form in (2.40). It is easy to check now that (2.40) has an optimal solution that has only one positive component (by selecting the method $m \sim q$ with the least coefficient). Therefore,

problem (2.40) is equivalent to

$$\min_{\mathbf{x}[q]} \langle \mathbf{c}, \mathbf{x}[q] \rangle + \sum_{i \in q, k \in K} \lambda_{ik} \left(\sum_{m:(i,k) \in m, m \sim q} x_m \right) \tag{2.41}$$

$$\text{s.t.} \quad \sum_{m:m \sim q} x_m = 1,$$

$$x_m \in \{0,1\}, \ \forall m \text{ such that } m \sim q.$$

Now, in turn, this minimization problem is equivalent to:

$$\min_{\{x^q_{ik}\}} \sum_{m:m \sim q} \left(\sum_{i \in q, k \in K} \frac{1}{u^q_{ik}} v^q_{ik} x^q_{ik} \mathbb{1}\{(i,k) \in m\} + \sum_{k \in K_m} \frac{1}{w^q_k} b^q_k \max_{i \in q} x^q_{ik} \right) + \tag{2.42}$$

$$+ \sum_{i \in q, k \in K} \lambda_{ik} \frac{1}{u^q_{ik}} \left(\sum_{m:m \sim q} \mathbb{1}\{(i,k) \in m\} \right) x^q_{ik} \tag{2.43}$$

$$\text{s.t.} \sum_{k \in K} x^q_{ik} = 1, \ \forall i \in q,$$

$$x^q_{ik} \in \{0,1\}, \ \forall i \in q, k \in K,$$

where v^q_{ik} and b^q_k are defined as in (2.37). Indeed, let $x_{m^*} = 1$ and $x_m = 0$ for all $m \neq m^*$ be the optimal solution to (2.41). Then, one can check that the feasible solution to problem (2.42) defined as $x^q_{ik} = 1$ for all $(i,k) \in m^*$ and $x^q_{ik} = 0$ otherwise, yields the same objective as (2.41). To see this, first note that, for each $k \in K_{m^*}$, there exists some $i \in q$ such that $(i,k) \in m^*$, i.e., $\max_{i \in q} x^q_{ik} = 1$. Thus, using the fact that $c_{m^*} = \sum_{i \in q, k \in K} v^q_{ik} \mathbb{1}\{(i,k) \in m^*\} + \sum_{k \in K_{m^*}} b^q_k$, and remembering that u^q_{ik} and w^q_k are used to avoid over-counting of costs, we see that the objective of (2.42) matches the one of (2.41). Therefore, the optimal value of (2.42) is less or equal than the optimal value of (2.41). We now prove the converse. In order to do that, first, note

that problem (2.42) can be rewritten as:

$$\min_{\{x_{ik}^q\}} \sum_{i\in q,k\in K} \frac{1}{u_{ik}^q}\left(\sum_{m:m\sim q} \mathbb{1}\{(i,k)\in m\}\right)(v_{ik}^q+\lambda_{ik})\,x_{ik}^q + \tag{2.44}$$

$$+\sum_{k\in K}\frac{1}{w_k^q}\left(\sum_{m:m\sim q}\mathbb{1}\{k\in K_m\}\right) b_k^q \max_{i\in q} x_{ik}^q \tag{2.45}$$

$$\text{s.t.} \sum_{k\in K} x_{ik}^q = 1,\ \forall i\in q,$$

$$x_{ik}^q\in\{0,1\},\ \forall i\in q, k\in K,$$

where the first term in the objective depends on x_{ik}^q and the second on $\max_{i\in q} x_{ik}^q$. Now, given the first constraint, we have that for any $i\in q$, there exists a unique $k_i\in K$ such that $x_{ik_i}^q=1$ and $x_{ik}^q=0$ for any $k\neq k_i$. Then, defining $m^*=\{(i,k_i): i\in q\}$, one can check that the feasible solution to problem (2.41) defined as $x_{m^*}=1$ and $x_m=0$ for all $m\neq m^*$, yields the same objective as (2.42).

Next, we develop an approximation algorithm through sequential randomized rounding to solve problem (2.44). For a comprehensive review of randomized rounding algorithms, we refer Williamson and Shmoys (2011) to the interested reader. Let us rewrite problem (2.44) (after linearizing the maximum) as follows:

$$\text{OPT}^q := \min \sum_{i\in q,k\in K} a_{ik}x_{ik}^q + \sum_{k\in K} b_k z_k^q \tag{2.46}$$

$$\text{s.t.} \sum_{k\in K} x_{ik}^q \geqslant 1,\ \forall i\in q, \tag{2.47}$$

$$z_k^q \geqslant x_{ik}^q,\ \forall i\in q, k\in K, \tag{2.48}$$

$$x_{ik}^q\in\{0,1\},\ \forall i\in q, k\in K, \tag{2.49}$$

for some $a_{ik}\geqslant 0$ for $i\in q, k\in K$ and $b_k\geqslant 0$ for $k\in K$. Relaxing the integer constraint

of problem (2.46), we obtain

$$V^q := \min \sum_{i \in q, k \in K} a_{ik} x^q_{ik} + \sum_{k \in K} b_k z^q_k \tag{2.50}$$

$$\text{s.t.} \sum_{k \in K} x^q_{ik} \geqslant 1, \ \forall i \in q, \tag{2.51}$$

$$z^q_k \geqslant x^q_{ik}, \ \forall i \in q, k \in K, \tag{2.52}$$

$$x^q_{ik} \in [0,1], \ \forall i \in q, k \in K, \tag{2.53}$$

so that, by definition, $V^q \leqslant \text{OPT}^q$. Then, we can prove the following.

Theorem 2.4. *For each fixed $q \in Q$, consider Algorithm 5 which takes an optimal basic feasible solution of* (2.50) *as input. Then, Algorithm 5 finds a solution that is feasible for* (2.46) *with probability at least $1 - 1/|q|$, and, for any constant $\theta > 2$, the cost of the feasible solution is at most $\theta 8 \log(|q|) OPT^q$ with probability at least $1 - 2/\theta$.*

Proof of Theorem 2.4: Let $q \in Q$ be given and consider iteration j of Algorithm 5. Define independent random variables $w^{(j)}_k \sim \text{Bernoulli}(z^{q*}_k)$ for each $k \in K$ and

$$y^{(j)}_{ik} = \begin{cases} w^{(j)}_k & \text{if } x^{q*}_{ik} \geqslant \frac{1}{2} z^{q*}_k; \\ 0 & \text{otherwise.} \end{cases}$$

Note that, for each $i \in q, k \in K$, the probability of violating constraints (2.48) is zero, while the probability of violating constraints (2.47) at iteration j can be upper bounded as follows:

$$\begin{aligned} \mathbb{P}\left(\sum_{k \in K} y^{(j)}_{ik} < 1 \right) &= \prod_{k \in K} (1 - \mathbb{P}(y^{(j)}_{ik} = 1)) \\ &\leqslant e^{-\sum_{k \in K} \mathbb{P}(y^{(j)}_{ik} = 1)} \\ &= e^{-\sum_{k \in K} z^{q*}_k \mathbb{1}\{x^{q*}_{ik} \geqslant \frac{1}{2} z^{q*}_k\}}. \end{aligned}$$

Algorithm 5 Sequential randomized rounding algorithm

Input: $(x_{ik}^{q*})_{i\in q,k\in K}$ and $(z_k^{q*})_{k\in K}$ optimal basic feasible solution to (2.50).

Define $\tilde{x}_{ik}^q = 0$, $\tilde{z}_k^q = 0$, for all $i \in q$, $k \in K$.

For $j = 1, \ldots, 4\log(|q|)$:

1. For $k \in K$, let $w_k^{(j)} = 1$ with probability z_k^{q*} and zero otherwise. Update $\tilde{z}_k^q = \tilde{z}_k^q + w_k^{(j)}$.

2. For $i \in q, k \in K$, let $y_{ik}^{(j)} = w_k^{(j)}$ if $x_{ik}^{q*} \geqslant \frac{1}{2} z_k^{q*}$ and zero otherwise. Update $\tilde{x}_{ik}^q = \tilde{x}_{ik}^q + y_{ik}^{(j)}$.

Update $\tilde{x}_{ik}^q = \min\{1, \tilde{x}_{ik}^q\}$, $\tilde{z}_k^q = \min\{1, \tilde{z}_k^q\}$, for all $i \in q$ $k \in K$.

Output: $(\tilde{x}_{ik}^q)_{i\in q,k\in K}$ and $(\tilde{z}_k^q)_{k\in K}$.

We now show that $\sum_{k\in K} z_k^{q*} \mathbb{1}\{x_{ik}^{q*} \geqslant \frac{1}{2} z_k^{q*}\} \geqslant \frac{1}{2}$ for any $i \in q$. First, note that if $z_k^{q*} = x_{ik}^{q*}$ for all k, then $\sum_{k\in K} z_k^{q*} \mathbb{1}\{x_{ik}^{q*} \geqslant \frac{1}{2} z_k^{q*}\} \geqslant 1$ (by constraint (2.51)) and so $\sum_{k\in K} z_k^{q*} \mathbb{1}\{x_{ik}^{q*} \geqslant \frac{1}{2} z_k^{q*}\} \geqslant \frac{1}{2}$ holds true. Next, we prove the following claim.

Claim 1: In problem (2.50), there exists a solution such that, for any $i \in q$, there exists at most one $k_i \in K$ such that $z_{k_i}^{q*} > x_{ik_i}^{q*} > 0$.

Proof of Claim 1: Suppose, by contradiction, that there exist k_i and k_i' such that $z_{k_i}^{q*} > x_{ik_i}^{q*} > 0$ and $z_{k_i'}^{q*} > x_{ik_i'}^{q*} > 0$. First, assume that $a_{ik_i} < a_{ik_i'}$. Then, for some $\epsilon > 0$, the feasible solution $x_{ik_i'}^{q*} - \epsilon$ and $x_{ik_i}^{q*} + \epsilon$ decreases the objective. But this is impossible, since $(x_{ik}^{q*})_{i\in q,k\in K}$ and $(z_k^{q*})_{k\in K}$ were assumed to be optimal (the case $a_{ik_i} > a_{ik_i'}$ can be proven similarly). Suppose now that $a_{ik_i} = a_{ik_i'}$. Then, for some $\epsilon > 0$, there are two feasible solutions $x_{ik_i'}^{q*} - \epsilon$, $x_{ik_i}^{q*} + \epsilon$, and $x_{ik_i'}^{q*} + \epsilon$, $x_{ik_i}^{q*} - \epsilon$, with the same objective. Thus, the feasible solution $(x_{ik}^{q*})_{i\in q,k\in K}$ can be rewritten as a convex combination of two other solutions, which is impossible because we assumed that our

solution is a basic feasible solution. This proves the claim. ■

Now, for any $i \in q$ such that there exists $k_i \in K$ with $z^*_{k_i} > x^{q*}_{ik_i}$, we have two cases:

1. If $x^{q*}_{ik_i} < \frac{1}{2}$, then $\sum_{k\in K} z^{q*}_k \mathbb{1}\{x^{q*}_{ik} \geqslant \frac{1}{2} z^{q*}_k\} \geqslant \sum_{k\neq k_i} z^{q*}_k \mathbb{1}\{x^{q*}_{ik} \geqslant \frac{1}{2} z^{q*}_k\} = \sum_{k\neq k_i} x^{q*}_{ik} > \frac{1}{2}$, where the last inequality holds because of constraint (2.51) and the fact that $x^{q*}_{ik_i} < \frac{1}{2}$.

2. If $x^{q*}_{ik_i} \geqslant \frac{1}{2}$, then $\sum_{k\in K} z^{q*}_k \mathbb{1}\{x^{q*}_{ik} \geqslant \frac{1}{2} z^{q*}_k\} \geqslant z^*_{k_i} \mathbb{1}\{x^{q*}_{ik_i} \geqslant \frac{1}{2} z^*_{k_i}\} = z^*_{k_i} > x^{q*}_{ik_i} \geqslant \frac{1}{2}$.

Therefore, we have that, for any $i \in q$, the probability of violating constraints (2.47) at iteration j can be upper bounded as follows:

$$\mathbb{P}\left(\sum_{k\in K} y^{(j)}_{ik} < 1\right) \leqslant \frac{1}{\sqrt{e}}.$$

According to Algorithm 5, we repeat the above process for $4\log(|q|)$ times. Therefore, for each $i \in q$, we have that

$$\mathbb{P}\left(\sum_{j=1}^{4\log(|q|)} \sum_{k\in K} y^{(j)}_{ik} < 1\right) = \mathbb{P}\left(\sum_{k\in K} y^{(j)}_{ik} < 1\right)^{4\log(|q|)} \leqslant \left(\frac{1}{\sqrt{e}}\right)^{4\log(|q|)} = \frac{1}{|q|^2}$$

Finally, the probability that Algorithm 5 finds a solution that is not feasible can be bounded as follows

$$\mathbb{P}\left(\exists i \in q : \sum_{j=1}^{4\log(|q|)} \sum_{k\in K} y^{(j)}_{ik} < 1\right) \leqslant \sum_{i\in q} \mathbb{P}\left(\sum_{j=1}^{4\log(|q|)} \sum_{k\in K} y^{(j)}_{ik} < 1\right) \leqslant \frac{|q|}{|q|^2} = \frac{1}{|q|},$$

i.e., the probability that Algorithm 5 finds a feasible solution is at least $1 - 1/|q|$. Now, note that

$$\begin{aligned}\mathbb{E}\left[\sum_{i\in q,k\in K} a_{ik} y^{(j)}_{ik} + \sum_{k\in K} b_k w^{(j)}_k\right] &= \sum_{i\in q,k\in K} a_{ik} z^{q*}_k \mathbb{1}\{x^{q*}_{ik} \geqslant \frac{1}{2} z^{q*}_k\} + \sum_{k\in K} b_k z^{q*}_k \\ &\leqslant 2\left[\sum_{i\in q,k\in K} a_{ik} x^{q*}_{ik} + \sum_{k\in K} b_k z^{q*}_k\right] = 2V^q \leqslant 2\mathrm{OPT}^q,\end{aligned}$$

and, moreover, the cost of the resulting solution returned by Algorithm 5 can be bounded by

$$\mathbb{E}\left[\sum_{i\in q,k\in K} a_{ik}\min\left\{1,\ \sum_{j=1}^{4\log(|q|)} y_{ik}^{(j)}\right\} + \sum_{k\in K} b_k \min\left\{1,\ \sum_{j=1}^{4\log(|q|)} w_k^{(j)}\right\}\right]$$

$$\leqslant \sum_{j=1}^{4\log(|q|)} \mathbb{E}\left[\sum_{i\in q,k\in K} a_{ik}y_{ik}^{(j)} + \sum_{k\in K} b_k w_k^{(j)}\right] \leqslant 4\log(|q|)2V^q \leqslant 8\log(|q|)\mathrm{OPT}^q.$$

Thus, by Markov inequality, we have that for some constant $\theta > 2$

$$\mathbb{P}\left(\sum_{i\in q,k\in K} a_{ik}\tilde{x}_{ik}^q + \sum_{k\in K} b_k\tilde{z}_k^q \geqslant \theta\cdot 8\log(|q|)\mathrm{OPT}^q\right) \leqslant \frac{1}{\theta},$$

and so, for a *feasible* solution, we have

$$\mathbb{P}\left(\sum_{i\in q,k\in K} a_{ik}\tilde{x}_{ik}^q + \sum_{k\in K} b_k\tilde{z}_k^q \geqslant \theta\cdot 8\log(|q|)\mathrm{OPT}^q \mid (\tilde{x}_{ik}^q)_{i\in q,k\in K},\ (\tilde{z}_k^q)_{k\in K}\ \text{feasible}\right)$$

$$\leqslant \frac{\mathbb{P}\left(\sum_{i\in q,k\in K} a_{ik}\tilde{x}_{ik}^q + \sum_{k\in K} b_k\tilde{z}_k^q \geqslant \theta\cdot 8\log(|q|)\mathrm{OPT}^q\right)}{\mathbb{P}((\tilde{x}_{ik}^q)_{i\in q,k\in K},\ (\tilde{z}_k^q)_{k\in K}\ \text{feasible})} \leqslant \frac{1/\theta}{1-1/|q|}.$$

Thus, because $|q| \geqslant 2$, we have that the probability that the cost of a feasible solution is greater than $\theta 8\log(|q|)\mathrm{OPT}^q$ is at most $2/\theta$. Therefore, the probability of obtaining a feasible solution with cost less than $\theta 8\log(|q|)\mathrm{OPT}^q$ is at least

$$\mathbb{P}\left(\sum_{i\in q,k\in K} a_{ik}\tilde{x}_{ik}^q + \sum_{k\in K} b_k\tilde{z}_k^q < \theta\cdot 8\log(|q|)\mathrm{OPT}^q \mid (\tilde{x}_{ik}^q)_{i\in q,k\in K},\ (\tilde{z}_k^q)_{k\in K}\ \text{feasible}\right)$$

$$= 1 - \mathbb{P}\left(\sum_{i\in q,k\in K} a_{ik}\tilde{x}_{ik}^q + \sum_{k\in K} b_k\tilde{z}_k^q \geqslant \theta\cdot 8\log(|q|)\mathrm{OPT}^q \mid (\tilde{x}_{ik}^q)_{i\in q,k\in K},\ (\tilde{z}_k^q)_{k\in K}\ \text{feasible}\right)$$

$$\geqslant 1 - \frac{2}{\theta}.$$

This completes the proof. ■

2.10.2 Proof of Proposition 2.3

In this section, we present a supergradient method for finding the primal solution of (2.10) when we only have access to an approximately optimal feasible solution (and not the optimal solution) of $L_q(\boldsymbol{\lambda})$. We then study the competitive ratio of our fulfillment strategy in this setting. Finally, we show that, using Theorem 2.4 from section 2.10.1, one can obtain a policy that is asymptotically $O(\log |q_{\max}|)$-competitive (as $s \to \infty$).

Proposition 2.5. *Denote by $L_q(\boldsymbol{\lambda}, \mathbf{x}[q])$ the objective of $L_q(\boldsymbol{\lambda})$ at $\mathbf{x}[q]$. Let $\mathbf{x}[q]^{(j)}$ be a randomized feasible solution to $L_q(\boldsymbol{\lambda}^{(j)})$ at iteration j of Algorithm 6 such that*

$$L_q(\boldsymbol{\lambda}^{(j)}, \mathbf{x}[q]^{(j)}) \leqslant \beta L_q(\boldsymbol{\lambda}^{(j)})$$

for some $\beta > 1$. Let C be a constant independent of J such that $\|G(\boldsymbol{\lambda}^{(j)})\|_2^2 \leqslant C \cdot OPT^E$ for all $j = 1, \ldots, J$ and let $\alpha_J = J^{-1/3}$. Denote by $\bar{\mathbf{x}} = \frac{1}{J}\sum_{j=1}^{J}\left(\sum_{q\in Q} p^a(q)\mathbf{x}[q]^{(j)}\right)$. Then, under Algorithm 6 we have that

$$\left\langle \mathbf{c},\ \beta^{-1}\bar{\mathbf{x}} \right\rangle \leqslant \left(1 + \frac{C}{2J^{1/3}}\right) OPT^E. \tag{2.54}$$

Moreover, for each $i \in I, k \in K$

$$\sum_{m:(i,k)\in m} \beta^{-1}\bar{x}_m - S_{ik} \leqslant \frac{\bar{C}}{J^{1/3}}, \tag{2.55}$$

for some positive constant $\bar{C}$ independent of J.

Proof of Proposition 2.5

Under Algorithm 6, we have that at iteration j

$$\begin{aligned}\|\boldsymbol{\lambda}^{(j+1)}\|_2^2 &\overset{(a)}{\leqslant} \|\boldsymbol{\lambda}^{(j)} + \alpha_J G(\boldsymbol{\lambda}^{(j)})\|_2^2 \\ &= \|\boldsymbol{\lambda}^{(j)}\|_2^2 + \alpha_J^2\|G(\boldsymbol{\lambda}^{(j)})\|_2^2 + 2\alpha_J\langle \boldsymbol{\lambda}^{(j)}, G(\boldsymbol{\lambda}^{(j)})\rangle,\end{aligned} \tag{2.56}$$

Algorithm 6 Projected supergradient method with fixed step-size

Input: Initialize $\boldsymbol{\lambda}^{(1)} = \mathbf{0}$, with $\mathbf{0} \in \mathbb{R}_{+}^{|I||K|}$, $\alpha_J \in [0,1]$ such that $\alpha_J J \to \infty$ and $\alpha_J \to 0$ as $J \to \infty$, β as in Proposition 2.5.

For $j = 1, \ldots, J$:

1. For each $q \in Q$, run Algorithm 5 until you find a *feasible solution* $\mathbf{x}[q]^{(j)} := \mathbf{x}^*[q](\boldsymbol{\lambda}^{(j)})$ for (2.40) whose cost is at most $\beta L_q(\boldsymbol{\lambda}^{(j)})$, where $\beta = 32\log(|q_{\max}|)$;
2. Choose $G(\boldsymbol{\lambda}^{(j)}) = \beta^{-1}\sum_{q\in Q} p^a(q) g(\mathbf{x}[q]^{(j)}) - \boldsymbol{S}$, with $g(\mathbf{x}[q]^{(j)}) = \left(\sum_{m:(i,k)\in m, m\sim q} x_m^{(j)}\right)_{i\in I, k\in K}$
3. Update $\boldsymbol{\lambda}^{(j+1)} = \max\{0, \boldsymbol{\lambda}^{(j)} + \alpha_J G(\boldsymbol{\lambda}^{(j)})\}$;

---

where (a) holds because of update rule $\boldsymbol{\lambda}^{(j+1)} = \max\{0, \boldsymbol{\lambda}^{(j)} + \alpha_J G(\boldsymbol{\lambda}^{(j)})\}$. Thus,

$$2\alpha_J \langle \boldsymbol{\lambda}^{(j)}, G(\boldsymbol{\lambda}^{(j)})\rangle \geqslant \|\boldsymbol{\lambda}^{(j+1)}\|_2^2 - \|\boldsymbol{\lambda}^{(j)}\|_2^2 - \alpha_J^2 \|G(\boldsymbol{\lambda}^{(j)})\|_2^2. \tag{2.57}$$

Moreover, note that at any iteration j, we have

$$\begin{aligned}
L(\boldsymbol{\lambda}^*) \geqslant L(\boldsymbol{\lambda}^{(j)}) &\overset{(a)}{=} \sum_{q\in Q} p^a(q) L_q(\boldsymbol{\lambda}^{(j)}) - \langle \boldsymbol{\lambda}^{(j)}, \boldsymbol{S}\rangle \\
&\overset{(b)}{\geqslant} \sum_{q\in Q} p^a(q)\beta^{-1} L_q(\boldsymbol{\lambda}^{(j)}, \mathbf{x}[q]^{(j)}) - \langle \boldsymbol{\lambda}^{(j)}, \boldsymbol{S}\rangle \\
&\overset{(c)}{=} \sum_{q\in Q} p^a(q)\beta^{-1} \left(\langle \mathbf{c}, \mathbf{x}[q]^{(j)}\rangle + \langle \boldsymbol{\lambda}^{(j)}, g(\mathbf{x}[q]^{(j)})\rangle\right) - \langle \boldsymbol{\lambda}^{(j)}, \boldsymbol{S}\rangle \\
&\overset{(d)}{=} \sum_{q\in Q} p^a(q)\beta^{-1} \left\langle \mathbf{c}, \mathbf{x}[q]^{(j)}\right\rangle + \left\langle \boldsymbol{\lambda}^{(j)}, G(\boldsymbol{\lambda}^{(j)})\right\rangle,
\end{aligned} \tag{2.58}$$

where (a) holds by the decomposition (2.20); (b) follows from our assumption on $L_q(\boldsymbol{\lambda}^{(j)})$; (c) holds by definition of $L_q(\boldsymbol{\lambda}^{(j)}, \mathbf{x}[q]^{(j)})$; and (d) follows from the definition of $G(\boldsymbol{\lambda}^{(j)})$ (in Algorithm 6). Thus, we have that at iteration j

$$L(\boldsymbol{\lambda}^*) - \sum_{q\in Q} p^a(q)\beta^{-1} \left\langle \mathbf{c}, \mathbf{x}[q]^{(j)}\right\rangle \geqslant \left\langle \boldsymbol{\lambda}^{(j)}, G(\boldsymbol{\lambda}^{(j)})\right\rangle. \tag{2.59}$$

This implies that at iteration j

$$L(\boldsymbol{\lambda}^*) - \sum_{q\in Q} p^a(q)\beta^{-1}\left\langle \mathbf{c}, \mathbf{x}[q]^{(j)}\right\rangle \geqslant \langle \boldsymbol{\lambda}^{(j)}, G(\boldsymbol{\lambda}^{(j)})\rangle$$

$$\overset{(a)}{\geqslant} \frac{1}{2\alpha_J}\left(\|\boldsymbol{\lambda}^{(j+1)}\|_2^2 - \|\boldsymbol{\lambda}^{(j)}\|_2^2 - \alpha_J^2\|G(\boldsymbol{\lambda}^{(j)})\|_2^2\right),$$

where (a) holds by using (2.57). Therefore, after averaging over all $j = 1, \ldots, J$ iterations and using the definition of $\bar{\mathbf{x}}$, we have that

$$\begin{aligned} OPT^E - \left\langle \mathbf{c},\ \beta^{-1}\bar{\mathbf{x}}\right\rangle &= L(\boldsymbol{\lambda}^*) - \left\langle \mathbf{c},\ \beta^{-1}\bar{\mathbf{x}}\right\rangle \\ &\geqslant \frac{1}{2J\alpha_J}\left(\|\boldsymbol{\lambda}^{(J+1)}\|_2^2 - \|\boldsymbol{\lambda}^{(1)}\|_2^2 - \alpha_J^2\sum_{j=1}^{J}\|G(\boldsymbol{\lambda}^{(j)})\|_2^2\right) \\ &\overset{(a)}{\geqslant} -\frac{C\cdot OPT^E}{2}\alpha_J, \end{aligned}$$

where (a) holds because $\boldsymbol{\lambda}^{(1)} = \mathbf{0}$ and by the assumption $\|G(\boldsymbol{\lambda}^{(j)})\|_2^2 \leqslant C\cdot OPT^E$. Rearranging the terms, we have, for $\alpha_J = J^{-1/3}$

$$\left\langle \mathbf{c},\ \beta^{-1}\bar{\mathbf{x}}\right\rangle \leqslant \left(1 + \frac{C}{2J^{1/3}}\right) OPT^E.$$

We now prove (2.55). First note that from (2.56)

$$\|\boldsymbol{\lambda}^{(j+1)}\|_2^2 \leqslant \|\boldsymbol{\lambda}^{(j)}\|_2^2 + \alpha_J^2\|G(\boldsymbol{\lambda}^{(j)})\|_2^2 + 2\alpha_J\langle \boldsymbol{\lambda}^{(j)}, G(\boldsymbol{\lambda}^{(j)})\rangle, \tag{2.60}$$

and, from (2.59), $OPT^E = L(\boldsymbol{\lambda}^*) \geqslant \sum_{q\in Q} p^a(q)\beta^{-1}\left\langle \mathbf{c}, \mathbf{x}[q]^{(j)}\right\rangle + \langle \boldsymbol{\lambda}^{(j)}, G(\boldsymbol{\lambda}^{(j)})\rangle$. Thus, for any j,

$$\langle \boldsymbol{\lambda}^{(j)}, G(\boldsymbol{\lambda}^{(j)})\rangle \leqslant OPT^E - \sum_{q\in Q} p^a(q)\beta^{-1}\left\langle \mathbf{c}, \mathbf{x}[q]^{(j)}\right\rangle \leqslant OPT^E.$$

Let $\hat{C} = OPT^E$. Since OPT^E is independent of the number of iterations J, $\langle \boldsymbol{\lambda}^{(j)}, G(\boldsymbol{\lambda}^{(j)})\rangle \leqslant \hat{C}$. Hence, from (2.60), we have that at iteration j

$$\|\boldsymbol{\lambda}^{(j+1)}\|_2^2 \leqslant \|\boldsymbol{\lambda}^{(j)}\|_2^2 + \alpha_J^2(C\cdot OPT^E) + 2\alpha_J\hat{C}. \tag{2.61}$$

Summing the previous inequality over all J iterations (and using the fact that $\boldsymbol{\lambda}^{(1)} = \mathbf{0}$), we obtain

$$\|\boldsymbol{\lambda}^{(J+1)}\|_2^2 \leqslant J\alpha_J^2(C \cdot OPT^E) + 2J\alpha_J\hat{C}.$$

In other terms, we have that

$$\frac{1}{J^2\alpha_J^2}\|\boldsymbol{\lambda}^{(J+1)}\|_2^2 \leqslant \frac{1}{J}(C \cdot OPT^E) + \frac{1}{J\alpha_J}2\hat{C}. \tag{2.62}$$

Now, note that at iteration j we have

$$\boldsymbol{\lambda}^{(j+1)} = \max\{0, \boldsymbol{\lambda}^{(j)} + \alpha_J G(\boldsymbol{\lambda}^{(j)})\} \geqslant \boldsymbol{\lambda}^{(j)} + \alpha_J G(\boldsymbol{\lambda}^{(j)}),$$

i.e.,

$$\boldsymbol{\lambda}^{(j+1)} - \boldsymbol{\lambda}^{(j)} \geqslant \alpha_J G(\boldsymbol{\lambda}^{(j)}).$$

Averaging the previous inequality over all J iterations (and using the fact that $\boldsymbol{\lambda}^{(1)} = 0$), we obtain

$$\frac{\boldsymbol{\lambda}^{(J+1)}}{J} \geqslant \alpha_J \frac{1}{J}\sum_{j=1}^{J} G(\boldsymbol{\lambda}^{(j)}) = \alpha_J\beta^{-1}\left(\frac{1}{J}\sum_{j=1}^{J}\sum_{q\in Q} p^a(q)g(\mathbf{x}[q]^{(j)})\right) - \boldsymbol{S}. \tag{2.63}$$

Moreover, note that for any $i \in I$ and $k \in K$,

$$\begin{aligned}\frac{1}{J}\sum_{j=1}^{J}\sum_{q\in Q} p^a(q)\left(\sum_{m:(i,k)\in m, m\sim q} x_m^{(j)}\right) &= \sum_{m:(i,k)\in m}\left(\frac{1}{J}\sum_{j=1}^{J}\sum_{q\in Q} p^a(q)x_m^{(j)}\mathbb{1}\{m \sim q\}\right)\\ &= \sum_{m:(i,k)\in m} \bar{x}_m,\end{aligned}$$

where the last equality holds by definition of $\bar{\mathbf{x}}$. Therefore, for any $i \in I$ and $k \in K$, we have that

$$\begin{aligned}\sum_{m:(i,k)\in m} \beta^{-1}\bar{x}_m - S_{ik} &\overset{(a)}{\leqslant} \frac{\lambda_{ik}^{(J+1)}}{J\alpha_J} \leqslant \frac{\|\boldsymbol{\lambda}^{(J+1)}\|_2}{J\alpha_J} \overset{(b)}{\leqslant} \sqrt{\frac{1}{J}(C \cdot OPT^E) + \frac{1}{J\alpha_J}2\hat{C}}\\ &\overset{(c)}{\leqslant} \frac{\bar{C}}{J^{1/3}},\end{aligned}$$

where (a) holds by using (2.63); (b) follows from (2.62); and (c) holds for J sufficiently large, $\alpha_J = J^{-1/3}$ and some positive constant $\bar{C}$ independent of J. This completes the proof. ■

Finally, we study the competitive ratio of our fulfillment strategy in this setting in which we only have access to an approximately optimal feasible solution of $L_q(\boldsymbol{\lambda}^{(j)})$ for each $q \in Q$, at each iteration j of Algorithm 6. We start by reminding the reader of some notation we use. Let

$$\bar{\mathbf{x}} = \frac{1}{J}\sum_{j=1}^{J}\sum_{q\in Q} p^a(q)\mathbf{x}[q]^{(j)} = \left(\frac{1}{J}\sum_{j=1}^{J}\sum_{q\in Q} p^a(q)x_m^{(j)}\mathbb{1}\{m \sim q\}\right)_{m\in M},$$

with each element denoted as $\bar{x}_m$, for $m \in M$. Then, $\bar{\mathbf{x}}$ satisfies constraint (2.11) of problem (2.10). Indeed, for $q' \in Q$

$$\begin{aligned}
\sum_{m:m\sim q'} \bar{x}_m &= \sum_{m:m\sim q'}\left(\frac{1}{J}\sum_{j=1}^{J}\sum_{q\in Q} p^a(q)x_m^{(j)}\mathbb{1}\{m \sim q\}\right)\\
&= \sum_{m\in M}\left(\frac{1}{J}\sum_{j=1}^{J}\sum_{q\in Q} p^a(q)x_m^{(j)}\mathbb{1}\{m \sim q\}\mathbb{1}\{m \sim q'\}\right)\\
&= \frac{1}{J}\sum_{j=1}^{J}\sum_{q\in Q} p^a(q)\left(\sum_{m\in M} x_m^{(j)}\mathbb{1}\{m \sim q\}\right)\mathbb{1}\{q = q'\}\\
&\overset{(a)}{=} \frac{1}{J}\sum_{j=1}^{J}\sum_{q\in Q} p^a(q)\mathbb{1}\{q = q'\}\\
&= p^a(q') \qquad (2.64)
\end{aligned}$$

where (a) holds because $\mathbf{x}[q]^{(j)}$ is a feasible solution to $L_q(\boldsymbol{\lambda}^{(j)})$ at iteration j of Algorithm 6 and so, by definition, $\sum_{m:m\sim q} x_m^{(j)} = 1$. Therefore, denoting by $\widetilde{\mathbf{x}} \coloneqq \beta^{-1}\bar{\mathbf{x}}$, with each element denoted as $\widetilde{x}_m$, we have that

$$\sum_{m:m\sim q'} \widetilde{x}_m = \beta^{-1}p^a(q'). \qquad (2.65)$$

We now show how to construct a solution for problem (2.10) which satisfies the inventory constraint. Let

$$\tau := \max\left\{1, \max_{i\in I, k\in K, S_{ik}>0} \frac{\sum_{m:(i,k)\in m} \widetilde{x}_m}{S_{ik}}\right\} \tag{2.66}$$

be its largest violation of the inventory constraint. For an order type $q \in Q$ and denoting by $m'(q)$ the discard method for q, define

$$y_m = \begin{cases} \widetilde{x}_m \cdot \tau^{-1} & \text{if } m \neq m'(q) \\ p^a(q) \cdot \left(1 + \tau^{-1} \frac{\widetilde{x}_{m'(q)} - \sum_{m:m\sim q} \widetilde{x}_m}{\sum_{m:m\sim q} \bar{x}_m}\right) & \text{if } m = m'(q). \end{cases} \tag{2.67}$$

Then, $\{y_m\}_{m\in M}$ is feasible for problem (2.10). Indeed, for a given $q \in Q$, we have that

$$\sum_{m:m\sim q} y_m = p^a(q).$$

Moreover,

$$\sum_{m:(i,k)\in m} y_m \overset{(a)}{=} \sum_{m:(i,k)\in m} \widetilde{x}_m \tau^{-1} \overset{(b)}{\leqslant} S_{ik},$$

where (a) holds because the sum does not involve the discard method $m'(q)$ (the inventory constraint is automatically satisfied for the dummy facility); and (b) follows from the definition of τ in (2.66).

Given this, next we state the main result of this section.

Theorem 2.5. *Suppose we use* $\mathbf{y} = \{y_m\}_{m\in M}$ *as defined in* (2.67) *to find a feasible solution to problem* (2.10) *in the offline process of Algorithm 1. Then, for any online multi-item fulfillment problem, when Algorithm 1 uses Algorithm 2 as a subroutine in Step 2 of the online process, we obtain a fulfillment strategy with a competitive ratio of at most*

$$\left(\frac{sJ^{1/3} + \kappa\beta\bar{C}}{sJ^{1/3} + \bar{C}} + \frac{\kappa(\beta-1)(sJ^{1/3})}{sJ^{1/3} + \bar{C}}\right)\left(1 + \frac{C}{2J^{1/3}}\right)\left(1 + \frac{(\kappa-1)|q_{\max}|}{\sqrt{s+3}}\right),$$

where J is the number of iterations in Algorithm 6 and $\bar{C}, C$ and β are as in Proposition 2.5.

Proof of Theorem 2.5: Consider the feasible solution $\mathbf{y} = \{y_m\}_{m \in M}$ as defined in (2.67). Suppose $\tau \neq 1$ (else, $\mathbf{y} = \widetilde{\mathbf{x}}$). Then, the expected cost of $\mathbf{y}$ can be bounded as follows:

$$\begin{aligned}
\langle \mathbf{c},\ \mathbf{y} \rangle &\overset{(a)}{=} \sum_{m:m\sim q} c_m \widetilde{x}_m \tau^{-1} + c_{m'(q)} \left(p^a(q) - \sum_{m:m\sim q} \widetilde{x}_m \tau^{-1} \right) \\
&\overset{(b)}{=} \sum_{m:m\sim q} c_m \widetilde{x}_m \tau^{-1} + c_{m'(q)} \left(\beta \sum_{m:m\sim q} \widetilde{x}_m - \sum_{m:m\sim q} \widetilde{x}_m \tau^{-1} \right) \\
&= \sum_{m:m\sim q} c_m \widetilde{x}_m \tau^{-1} + \sum_{m:m\sim q} c_{m'(q)} \left(\beta \widetilde{x}_m - \widetilde{x}_m \tau^{-1} \right) \\
&\overset{(c)}{\leqslant} \sum_{m:m\sim q} c_m \widetilde{x}_m \tau^{-1} + \sum_{m:m\sim q} \kappa c_m \left(\beta \widetilde{x}_m - \widetilde{x}_m \tau^{-1} \right) \\
&= \sum_{m:m\sim q} c_m \widetilde{x}_m \cdot \left((1-\kappa)\tau^{-1} + \beta\kappa \right) \\
&\overset{(d)}{\leqslant} \sum_{m:m\sim q} c_m \widetilde{x}_m \cdot \frac{sJ^{1/3} + \beta\kappa(sJ^{1/3} + \bar{C}) - \kappa s J^{1/3}}{sJ^{1/3} + \bar{C}} \\
&= \sum_{m:m\sim q} c_m \widetilde{x}_m \cdot \left(\frac{sJ^{1/3} + \kappa\beta\bar{C}}{sJ^{1/3} + \bar{C}} + \frac{\kappa(\beta-1)(sJ^{1/3})}{sJ^{1/3} + \bar{C}} \right) \\
&= \left(\frac{sJ^{1/3} + \kappa\beta\bar{C}}{sJ^{1/3} + \bar{C}} + \frac{\kappa(\beta-1)(sJ^{1/3})}{sJ^{1/3} + \bar{C}} \right) \langle \mathbf{c}, \widetilde{\mathbf{x}} \rangle
\end{aligned}$$

where (a) holds by using definition (2.67) and the fact that $\sum_{m:m\sim q} \bar{x}_m = p^a(q)$; (b) holds by using (2.65); (c) follows from the assumption that for some $\kappa > 1$, $c_{m'(q)}/c_m \leqslant \kappa$, for all $m \sim q$ and all $q \in Q$; and (d) holds from the fact that $(1-k) < 0$ and because, from (2.55), we have that $\tau \leqslant 1 + \bar{C}/(sJ^{1/3})$, with $s = \min_{i\in I, k\in K, S_{ik}>0}\{S_{ik}\}$

Therefore, from (2.54), we have that

$$\begin{aligned}\langle \mathbf{c},\ \mathbf{y}\rangle &\leqslant \left(\frac{sJ^{1/3}+\kappa\beta\bar{C}}{sJ^{1/3}+\bar{C}}+\frac{\kappa(\beta-1)(sJ^{1/3})}{sJ^{1/3}+\bar{C}}\right)\langle \mathbf{c},\widetilde{\mathbf{x}}\rangle \\ &\leqslant \left(\frac{sJ^{1/3}+\kappa\beta\bar{C}}{sJ^{1/3}+\bar{C}}+\frac{\kappa(\beta-1)(sJ^{1/3})}{sJ^{1/3}+\bar{C}}\right)\left(1+\frac{C}{2J^{1/3}}\right)OPT^{E}.\end{aligned}$$

Now, following similar steps to the proof of Lemma 2.3 and by Proposition 2.1, we have that when Algorithm 1 uses Algorithm 2 as a subroutine in Step 2 of the online process achieves a competitive ratio of at most

$$\left(\frac{sJ^{1/3}+\kappa\beta\bar{C}}{sJ^{1/3}+\bar{C}}+\frac{\kappa(\beta-1)(sJ^{1/3})}{sJ^{1/3}+\bar{C}}\right)\left(1+\frac{C}{2J^{1/3}}\right)\left(1+\frac{(\kappa-1)|q_{\max}|}{\sqrt{s+3}}\right). \tag{2.68}$$

This completes the proof. ■

Finally, note that using Theorem 2.4 (from Section 2.10.1) with $\theta = 4$ to approximately solve $L_q(\boldsymbol{\lambda}^{(j)})$, we have that at iteration j in Algorithm 6

$$L_q(\boldsymbol{\lambda}^{(j)}, \mathbf{x}[q]^{(j)}) \leqslant 32\log(|q_{\max}|)L_q(\boldsymbol{\lambda}^{(j)}).$$

Therefore, the assumption of Proposition 2.5 holds with $\beta = 32\log(|q_{\max}|)$. Moreover, since

$$\lim_{s\to\infty}\left(\frac{sJ^{1/3}+\kappa\beta\bar{C}}{sJ^{1/3}+\bar{C}}+\frac{\kappa(\beta-1)(sJ^{1/3})}{sJ^{1/3}+\bar{C}}\right) = 1+\kappa(\beta-1),$$

we have that, from Theorem 2.5, our fulfillment strategy is asymptotically $O(\log|q_{\max}|)$-competitive, as $s\to\infty$.

It is worth mentioning that both Algorithm 6 and Ma (2023) are essentially finding $O(\log|q_{\max}|)$-optimal solutions for (2.10). Moreover, in terms of the number of LPs to solve, assuming that every order contains n items, the algorithm of Ma (2023) solves one large LP with $|Q||K|n + |Q||K|$ variables, while Algorithm 6 solves in expectation at most $2J|Q|$ LPs, where each LP has $|K|n + |K|$ variables. Specifically, because according to Algorithm 6, for each $q \in Q$, we run Algorithm 5 until we find a

feasible solution $\mathbf{x}[q]^{(j)}$, by choosing $\theta = 4$ in Theorem 2.4, we have that the expected number of iterations of step 1 in Algorithm 6 is at most 2. Thus, the algorithm of Ma (2023) has the computational advantage of solving one single LP when $|Q|$ is not too large. We also note that if the LP solved in the algorithm of Ma (2023) becomes too large, it can be solved through a Lagrangian relaxation technique similar to the one we develop in Algorithm 6.

2.11 Details of Numerical Simulations

In this section, we provide details about our simulation. We start by initializing several parameters: the number of items $|I|$ is set to 20, the number of facilities is $|K| = 5$, the number of cities is $|J| = 10$ and $T = 1000$. Note that we follow a simulation environment as close as possible to Ma (2023) and Jasin and Sinha (2015), and details of our simulation environment, including data and code can be found in Amil et al. (2022).

2.11.1 Facilities, Cities, and Costs

Similarly to Ma (2023), we consider an e-commerce network with the 10 biggest U.S. cities that are geographically spread across the country and the 5 largest facility centers that are centrally located (details can be found in Amil et al. (2022)). We generate 30 instances for this network, where each instance differs by order types, demand rates, and initial inventories. For each instance, we draw 100 sequences of order arrivals of length $T = 1000$.

Following Jasin and Sinha (2015) and Ma (2023), we assume that each item is exactly one pound and the fixed cost of shipping anything from any warehouse is 8.759. Moreover, for each facility k and city j, the variable cost of shipping an item from facility k to city j is calculated as $0.423 + 0.000541 * \texttt{dists[k][j]}$, where `dists[k][j]` is the haversine distance in miles between facility k and city j (see

Jasin and Sinha (2015) for details). The cost of not fulfilling an item is assumed to be twice as much as the maximum cost of fulfilling it (i.e., a factor of 2 times the maximum distance between any facility k and city j). Therefore, adopting the assumption that the lost-sale penalty cost is 2, we can estimate the value for κ in our simulation. Remember that, in our model, κ is defined as the maximum ratio between the cost of not fulfilling an order and the cost of using any other method for that order, i.e., $\kappa = \max_{q \in Q} \frac{c_{\max}(q)}{\min_{m \sim q} c_m}$. Hence, using this definition, the estimated average value of κ in our simulation (over 30 different instances of our e-commerce network) is as follows: when the maximum order size is 2, $\kappa_{\text{avg}} = 4.34$; when the maximum order size is 5, $\kappa_{\text{avg}} = 9.34$.

2.11.2 Order Types, Demand Rates, Inventories and Methods

Following Jasin and Sinha (2015) and Ma (2023), we consider orders of different sizes. Specifically, for each order size $n = 1, \ldots, n_{\max}$, we uniformly generate n_0 distinct orders. Size $n = 0$ is also generated. We consider $n_{\max} \in \{2, 5\}$ and $n_0 = 5$. Each unique combination of n items is selected randomly to match the specified size from the total item set I. Moreover, note that, as described in our model, each order type is characterized by the items in the order and the location or city j from which it is ordered. Thus, for each given combination of items, we generate $|J|$ orders (i.e. same combination of items, with $|J|$ different locations). We therefore generate a total of $|J|(1 + n_{\max} n_0)$ order types.

Demand rates.

In each period $t = 1, \ldots, T$, exactly one location-specific order arrives. The arrival probabilities are determined as follows. First, we randomly generate $p(n)$: the probability that an order of size n arrives. These probabilities are such that $\sum_{n=0}^{n_{\max}} p(n) = 1$. Second, for each size $n \in \{1, \ldots, n_{\max}\}$, we randomly generate $p(n, m)$:

the probability that an order of size n is of combination m, for $m = 1, \ldots, n_0$. These probabilities are such that $\sum_{m=1}^{n_0} p(n, m) = p(n)$. Finally, for each size n and combination m, we scale the probabilities by the population of location j, obtaining $p(n, m, j) \coloneqq p(n, m)\frac{\text{population}(j)}{\text{total population}}$, where population$(j)$ is the population from city j and total population is the total population from all cities. Note that $p(n, m, j)$ is the probability that an order of size n, combination m, arrives from location j. These probabilities are such that $\sum_{j=1}^{|J|} p(n, m, j) = p(n, m)$.

Inventories.

Each facility k stocks item i independently with probability $p_{\text{stock}} = 0.75$, i.e., for each pair of facility k and item i we determine whether k holds item i using a Bernoulli random variable with probability of success p_{stock}. Then, for each region j, we find the closest facility k that holds item i. Then, for an item i, we calculate its expected demand at facility k, i.e. $\text{demand}_{ik} = \sum_{j=1}^{|J|} \sum_{(n,q):i \in q} \mathbb{1}\{\text{closest}_{ij} = k\} p(n, q, j)$, where the sum is over all cities and orders containing item i. Note that, for item $i \in q$, we consider only regions j for which facility k is the closest when summing over the arrival probabilities. Finally, inventories for pair (i, k) are set as $S_{ik} = \lceil T\text{demand}_{ik} \rceil$.

Methods.

We closely follow the definition of methods provided in Section 2.2, i.e., a method is defined as a set of item-facility pairs. First, for each item i in the order, we find the set of facilities K_i from which the item can be fulfilled. Then, we list all methods m that can be used to fulfill the order (including partial fulfillment and no fulfillment).

Considering the above setting, we run our policy described in Remark 2.1, the state-of-the-art correlated rounding policy of Ma (2023) and compare the costs and LPs in Tables 2.2 and 2.3, respectively.

2.12 Extension of our Main Result

In this section, we show an extension of the main result of this chapter. Specifically, we show that we can improve the bound provided in Theorem 2.1 to have the expectation of $|q|$ instead of its maximum.

2.12.1 A Competitive Ratio Based on the Average Order Size

Here, we provide the proof of the competitive ratio that depends on the average order size instead of the maximum order size. The formal statement is provided next.

Theorem 2.6. *For any online multi-item fulfillment problem, when Algorithm 1 uses Algorithm 2 as a subroutine in Step 2 of the online process, we obtain a fulfillment strategy with a competitive ratio of at most*

$$1 + \frac{(\kappa - 1)\mathbb{E}_{q\sim F(\cdot)}[|q|]}{\sqrt{s+3}}, \tag{2.69}$$

where $s = \min_{i\in I, k\in K}\{S_{ik}\}$ *and the probability mass function* $F(q)$ *is given by*

$$F(q) = \frac{\sum_{t\in[T]}\sum_{m\in M(q)} c_m y_m^t}{\sum_{q'\in Q}\left(\sum_{t\in[T]}\sum_{m\in M(q')} c_m y_m^t\right)}.$$

Before proving this extension, we highlight that $F(q)$ can be interpreted as the proportion of optimal cost incurred by the decision to fulfill order type q, in the same spirit of Jasin and Sinha (2015).

Proof: The proof mirrors those stated in the previous sections. We next outline the extra steps required to obtain the competitive ratio.

Following similar steps to the proof of Proposition 2.1, we have that

$$\begin{aligned}&\mathbb{P}(\text{accepting the method at } t \mid \text{the method is of type } m \text{ and } q \text{ arrives})\\ &\geqslant \gamma(|q|) = 1 - \frac{|q|}{\sqrt{s+3}},\end{aligned}$$

Now, following similar steps to the proof of Lemma 2.2, we have that the expected cost at time t is:

$$\sum_{q\in Q}\gamma(|q|)\left(\sum_{m\in M(q)}c_m\frac{y_m^t}{p_t(q)}\right)p_t(q)+\sum_{q\in Q}(1-\gamma(|q|))\left(\sum_{m\in M(q)}c_{m'(q)}\frac{y_m^t}{p_t(q)}\right)p_t(q)$$

$$\overset{(a)}{\leqslant}\mathbb{E}_{q\sim p_t(\cdot)}[\gamma(|q|)A_t(q)+(1-\gamma(|q|))\kappa A_t(q)]$$

$$=\mathbb{E}_{q\sim p_t(\cdot)}[A_t(q)+(\kappa-1)(1-\gamma(|q|))A_t(q)],$$

where

$$A_t(q)=\sum_{m\in M(q)}c_m\frac{y_m^t}{p_t(q)}\text{ and }B_t(q)=\sum_{m\in M(q)}c_{m'(q)}\frac{y_m^t}{p_t(q)},$$

and (a) follows by noting that $B_t(q)\leqslant\kappa A_t(q)$. Thus, the competitive ratio of our fulfillment policy can be upper bounded by

$$\frac{\sum_{t=1}^T\mathbb{E}_{q\sim p_t(\cdot)}[A_t(q)+(\kappa-1)(1-\gamma(|q|))A_t(q)]}{\sum_{t=1}^T\mathbb{E}_{q\sim p_t(\cdot)}[A_t(q)]}$$

$$=1+(\kappa-1)\frac{1}{\sqrt{s+3}}\frac{\mathbb{E}_{q\sim p_t(\cdot)}[|q|\sum_{t=1}^T A_t(q)]}{\mathbb{E}_{q\sim p_t(\cdot)}[\sum_{t=1}^T A_t(q)]},$$

where the equality holds by substituting the expression for $\gamma(|q|)$. Note that by using the probability mass function defined in the theorem statement, the above upper bound on the competitive ratio becomes

$$1+\frac{(\kappa-1)\mathbb{E}_{q\sim F(\cdot)}[|q|]}{\sqrt{s+3}}.$$

This completes the proof. ■

2.13 Additional Proofs

Proof of Lemma 2.1:

Let $\mathbf{y}(\mathcal{D})=\{y_m^t(\mathcal{D})\}_{m\in M,t\in[T]}$ be the optimal solution to (2.1). Thus, it must satisfy $\sum_{m:m\sim q}y_m^t(\mathcal{D})=\mathcal{D}^{qt},\ \forall q\in Q,t\in[T]$, and $\sum_{t\in[T]}\sum_{m:(i,k)\in m}y_m^t(\mathcal{D})\leqslant S_{ik},\ \forall i\in I,k\in$

K. Taking expectation of both sides of these two equations and summing over t for the first constraint we obtain

$$\sum_{m:m\sim q} \mathbb{E}\left[\sum_{t\in[T]} y_m^t(\mathcal{D})\right] = \sum_{t\in[T]} p_t(q) = p^a(q),\ \forall q \in Q,$$

and

$$\sum_{m:(i,k)\in m} \mathbb{E}\left[\sum_{t\in[T]} y_m^t(\mathcal{D})\right] \leqslant S_{ik},\ \forall i \in I, k \in K.$$

Note also that $\mathbb{E}\left[\sum_{t\in[T]} y_m^t(\mathcal{D})\right] \in [0,T]$, i.e., $\left\{\mathbb{E}\left[\sum_{t\in[T]} y_m^t(\mathcal{D})\right]\right\}_{m\in M}$ is a feasible solution to problem (2.10). Therefore, by definition,

$$\begin{aligned}
\mathrm{OPT}^E &\leqslant \sum_{q\in Q}\sum_{m:m\sim q} c_m \mathbb{E}\left[\sum_{t\in[T]} y_m^t(\mathcal{D})\right] \\
&= \mathbb{E}\left[\sum_{t\in[T]}\sum_{q\in Q}\sum_{m:m\sim q} c_m y_m^t(\mathcal{D})\right] \\
&= \mathbb{E}\left[\mathrm{OPT}(\mathcal{D})\right].
\end{aligned}$$

This completes the proof. ■

Proof of Lemma 2.2:

Following Algorithm 1, define $m'(q)$ as the *discard method*, i.e., using the dummy facility to satisfy all the items in q. Remember that, without loss of generality, we assume that the cost of $m'(q)$ is the largest among all $m \in M(q)$ for each order type q.

Denote by $O^t = \{\text{accept the method at time } t\}$ the event in which we accept the method at time t and by $C_m^t = \{\text{method at time } t \text{ is type } m\}$ be the event in which method m is drawn for fulfillment at time t. Then, according to Algorithm 1, we

have that the expected cost at time t is

$$\sum_{q\in Q}\left[\sum_{m\in M(q)}\frac{y_m^t}{p_t(q)}\left(c_m\mathbb{P}(O^t \mid C_m^t, \mathcal{D}^{qt}=1)+c_{m'(q)}(1-\mathbb{P}(O^t \mid C_m^t, \mathcal{D}^{qt}=1))\right)\right]p_t(q)$$

$$=\sum_{q\in Q}\left[\sum_{m\in M(q)}\frac{y_m^t}{p_t(q)}\left((c_m-c_{m'(q)})\mathbb{P}(O^t \mid C_m^t, \mathcal{D}^{qt}=1)+c_{m'(q)}\right)\right]p_t(q)$$

$$\overset{(a)}{\leqslant}\sum_{q\in Q}\left[\sum_{m\in M(q)}\frac{y_m^t}{p_t(q)}\left((c_m-c_{m'(q)})\gamma+c_{m'(q)}\right)\right]p_t(q)$$

$$=\sum_{q\in Q}\left[\sum_{m\in M(q)}\frac{y_m^t}{p_t(q)}\left(\gamma c_m+c_{m'(q)}(1-\gamma)\right)\right]p_t(q)$$

$$=\gamma\sum_{q\in Q}\sum_{m\in M(q)}c_m y_m^t+(1-\gamma)\sum_{q\in Q}\sum_{m\in M(q)}c_{m'(q)}y_m^t,$$

where (a) holds by definition of γ-conservative method-acceptance strategy and the fact that $c_m \leqslant c_{m'(q)}$. Therefore, using $\mathbb{E}\left[\mathrm{ALG}_1(\mathcal{D})\right]$ to denote the expected cost incurred by Algorithm 1, we have that

$$\frac{\mathbb{E}\left[\mathrm{ALG}_1(\mathcal{D})\right]}{\mathbb{E}\left[\mathrm{OPT}(\mathcal{D})\right]}\overset{(a)}{\leqslant}\frac{\mathbb{E}\left[\mathrm{ALG}_1(\mathcal{D})\right]}{\sum_{t\in[T]}\sum_{q\in Q}\sum_{m:m\sim q}c_m y_m^t}$$

$$\leqslant\frac{\gamma\sum_{t\in[T]}\sum_{q\in Q}\sum_{m\in M(q)}c_m y_m^t+(1-\gamma)\sum_{t\in[T]}\sum_{q\in Q}\sum_{m\in M(q)}c_{m'(q)}y_m^t}{\sum_{t\in[T]}\sum_{q\in Q}\sum_{m:m\sim q}c_m y_m^t}$$

$$\leqslant\gamma+(1-\gamma)\frac{\sum_{t\in[T]}\sum_{q\in Q}\sum_{m\in M(q)}c_{m'(q)}y_m^t}{\sum_{t\in[T]}\sum_{q\in Q}\sum_{m:m\sim q}c_m y_m^t},$$

where (a) holds by using Lemma 2.1. Thus, since for some $\kappa>1$, $c_{m'(q)}/c_m \leqslant \kappa$, for all $m \in M(q)$ and any q, we have that

$$\frac{\mathbb{E}\left[\mathrm{ALG}_1(\mathcal{D})\right]}{\mathbb{E}\left[\mathrm{OPT}(\mathcal{D})\right]}\leqslant\gamma+(1-\gamma)\frac{\kappa\sum_{t\in[T]}\sum_{q\in Q}\sum_{m\in M(q)}c_m y_m^t}{\sum_{t\in[T]}\sum_{q\in Q}\sum_{m:m\sim q}c_m y_m^t}=1+(\kappa-1)(1-\gamma).$$

This completes the proof. ■

Proof of Lemma 2.3:

Let $\{\mathbf{y}_m\}_{m\in M}$ be an optimal solution to the relaxation (2.10) and let $\{\hat{\mathbf{y}}_m\}_{m\in M}$ be an ϵ-approximation solution to (2.10). Then, we have that

$$\begin{aligned}
\frac{\mathbb{E}\left[\mathrm{ALG}_1(\mathcal{D})\right]}{\mathbb{E}\left[\mathrm{OPT}(\mathcal{D})\right]} &\overset{(a)}{\leqslant} \frac{\mathbb{E}\left[\mathrm{ALG}_1(\mathcal{D})\right]}{\sum_{q\in Q}\sum_{m:m\sim q} c_m y_m} \\
&\overset{(b)}{\leqslant} \frac{\gamma\sum_{q\in Q}\sum_{m\in M(q)} c_m \hat{y}_m + (1-\gamma)\kappa\sum_{q\in Q}\sum_{m\in M(q)} c_m \hat{y}_m}{\sum_{q\in Q}\sum_{m:m\sim q} c_m y_m} \\
&\overset{(c)}{\leqslant} \frac{(1+\epsilon)(1+(\kappa-1)(1-\gamma))\sum_{q\in Q}\sum_{m:m\sim q} c_m y_m}{\sum_{q\in Q}\sum_{m:m\sim q} c_m y_m} \\
&= (1+\epsilon)(1+(\kappa-1)(1-\gamma)),
\end{aligned}$$

where (a) follows from invoking Lemma 2.1, (b) holds by following similar steps we used in the proof of Lemma 2.2 (using the ϵ-approximation $\{\hat{\mathbf{y}}_m\}_{m\in M}$), and (c) follows from the definition of ϵ-approximation. ■

Proof of Lemma 2.4:

We want to prove that

$$L_q(\mu) \leqslant L_q(\boldsymbol{\lambda}) + \langle g_q(\boldsymbol{\lambda}), (\mu - \boldsymbol{\lambda})\rangle \quad \text{for all } \mu \in \mathbb{R}_+^{|I||K|}.$$

Note that, by definition of $g_q(\cdot)$,

$$\begin{aligned}
L_q(\boldsymbol{\lambda}) + \langle g_q(\boldsymbol{\lambda}), (\mu - \boldsymbol{\lambda})\rangle &\overset{(a)}{=} \langle \mathbf{c}, \mathbf{x}^*[q](\boldsymbol{\lambda})\rangle + \langle \boldsymbol{\lambda}, g_q(\boldsymbol{\lambda})\rangle + \langle g_q(\boldsymbol{\lambda}), (\mu - \boldsymbol{\lambda})\rangle \\
&= \langle \mathbf{c}, \mathbf{x}^*[q](\boldsymbol{\lambda})\rangle + \langle \mu, g_q(\boldsymbol{\lambda})\rangle \\
&\overset{(b)}{\geqslant} \langle \mathbf{c}, \mathbf{x}^*[q](\mu)\rangle + \langle \mu, g_q(\mu)\rangle \\
&= L_q(\mu),
\end{aligned}$$

where (a) holds because $\langle \boldsymbol{\lambda}, g_q(\boldsymbol{\lambda}) \rangle = \sum_{i \in q, k \in K} \lambda_{ik} \left(\sum_{m:(i,k) \in m, m \sim q} x^*_m(\boldsymbol{\lambda}) \right)$ and inequality (b) holds because, for $\mu \in \mathbb{R}_+^{|I||K|}$, the optimum of $L_q(\cdot)$ is achieved at $\mathbf{x}^*[q](\mu)$. This completes the proof. ■

Proof of Proposition 2.2:

Under Algorithm 3, at each iteration j, we have that

$$\begin{aligned}
\|\boldsymbol{\lambda}^{(j+1)} - \boldsymbol{\lambda}^*\|_2^2 &\overset{(a)}{\leqslant} \|\boldsymbol{\lambda}^{(j)} + \alpha_J G(\boldsymbol{\lambda}^{(j)}) - \boldsymbol{\lambda}^*\|_2^2 \\
&= \|\boldsymbol{\lambda}^{(j)} - \boldsymbol{\lambda}^*\|_2^2 + 2\alpha_J \langle G(\boldsymbol{\lambda}^{(j)}), (\boldsymbol{\lambda}^{(j)} - \boldsymbol{\lambda}^*) \rangle + \alpha_J^2 \|G(\boldsymbol{\lambda}^{(j)})\|_2^2 \\
&\overset{(b)}{\leqslant} \|\boldsymbol{\lambda}^{(j)} - \boldsymbol{\lambda}^*\|_2^2 + 2\alpha_J (L(\boldsymbol{\lambda}^{(j)}) - L(\boldsymbol{\lambda}^*)) + \alpha_J^2 \|G(\boldsymbol{\lambda}^{(j)})\|_2^2,
\end{aligned}$$

where inequality (a) holds because the projection operator onto $\mathbb{R}_+^{|I||K|}$ is nonexpansive and (b) holds since $G(\boldsymbol{\lambda}^{(j)})$ is a supergradient of $L(\boldsymbol{\lambda}^{(j)})$ at $\boldsymbol{\lambda}^{(j)}$. Therefore, iteratively applying the previous inequality, we obtain

$$\|\boldsymbol{\lambda}^{(j+1)} - \boldsymbol{\lambda}^*\|_2^2 \leqslant \|\boldsymbol{\lambda}^{(1)} - \boldsymbol{\lambda}^*\|_2^2 + 2\alpha_J \sum_{i=1}^{j} (L(\boldsymbol{\lambda}^{(i)}) - L(\boldsymbol{\lambda}^*)) + \alpha_J^2 \sum_{i=1}^{j} \|G(\boldsymbol{\lambda}^{(i)})\|_2^2.$$

Now, note that, since $\|\boldsymbol{\lambda}^{(j+1)} - \boldsymbol{\lambda}^*\|_2^2 \geqslant 0$, we have

$$\begin{aligned}
2\alpha_J \sum_{i=1}^{j} (L(\boldsymbol{\lambda}^*) - L(\boldsymbol{\lambda}^{(i)})) &\leqslant \|\boldsymbol{\lambda}^{(1)} - \boldsymbol{\lambda}^*\|_2^2 + \alpha_J^2 \sum_{i=1}^{j} \|G(\boldsymbol{\lambda}^{(i)})\|_2^2 \\
&\leqslant \|\boldsymbol{\lambda}^{(1)} - \boldsymbol{\lambda}^*\|_2^2 + \alpha_J^2 j (C \cdot OPT^E). \qquad (2.70)
\end{aligned}$$

Moreover, at iteration J, we have

$$\begin{aligned}\sum_{i=1}^{J}\alpha_J(L(\boldsymbol{\lambda}^*)-L(\boldsymbol{\lambda}^{(i)})) &= L(\boldsymbol{\lambda}^*)\left(\sum_{i=1}^{J}\alpha_J\right)-\alpha_J\sum_{i=1}^{J}L(\boldsymbol{\lambda}^{(i)})\\ &\overset{(a)}{\geqslant} J\alpha_J L(\boldsymbol{\lambda}^*)-J\alpha_J L\left(\frac{1}{J}\sum_{i=1}^{J}\boldsymbol{\lambda}^{(i)}\right)\\ &= J\alpha_J\left(L(\boldsymbol{\lambda}^*)-L\left(\frac{1}{J}\sum_{i=1}^{J}\boldsymbol{\lambda}^{(i)}\right)\right)\\ &= \left(L^*-L_{\text{avg}}^{(J)}\right)\left(J\alpha_J\right),\end{aligned}$$

where (a) holds by concavity of $L(\cdot)$. Therefore, by using (2.70) for $j=J$, we have that

$$L^*-L_{\text{avg}}^{(J)}\leqslant\frac{\|\boldsymbol{\lambda}^{(1)}-\boldsymbol{\lambda}^*\|_2^2+\alpha_J^2J(C\cdot OPT^E)}{2\alpha_JJ}=\frac{\|\boldsymbol{\lambda}^*\|_2^2+\alpha_J^2J(C\cdot OPT^E)}{2\alpha_JJ}.$$

Finally, note that the constant C always exists. Indeed, we can find an explicit expression in terms of the primitives of the model by upper bounding $\|G(\boldsymbol{\lambda}^{(j)})\|_2^2$. Specifically, whenever

$$C\geqslant\frac{|q_{\max}|T^2+S_{\max}^2|I||K|}{OPT^E},$$

where $S_{\max}=\max_{i\in I,k\in K}\{S_{ik}\}$, we have that $\|G(\boldsymbol{\lambda}^{(j)})\|_2^2\leqslant C\cdot OPT^E$. This completes the proof. ■

Proof of Theorem 2.2:

Under Algorithm 3, we have that at iteration j

$$\begin{aligned}\|\boldsymbol{\lambda}^{(j+1)}\|_2^2 &\overset{(a)}{\leqslant} \|\boldsymbol{\lambda}^{(j)}+\alpha_JG(\boldsymbol{\lambda}^{(j)})\|_2^2\\ &= \|\boldsymbol{\lambda}^{(j)}\|_2^2+\alpha_J^2\|G(\boldsymbol{\lambda}^{(j)})\|_2^2+2\alpha_J\langle\boldsymbol{\lambda}^{(j)},G(\boldsymbol{\lambda}^{(j)})\rangle,\end{aligned}\tag{2.71}$$

where (a) holds because of the update rule $\boldsymbol{\lambda}^{(j+1)}=\max\{0,\boldsymbol{\lambda}^{(j)}+\alpha_JG(\boldsymbol{\lambda}^{(j)})\}$. Thus,

$$2\alpha_J\langle\boldsymbol{\lambda}^{(j)},G(\boldsymbol{\lambda}^{(j)})\rangle\geqslant\|\boldsymbol{\lambda}^{(j+1)}\|_2^2-\|\boldsymbol{\lambda}^{(j)}\|_2^2-\alpha_J^2\|G(\boldsymbol{\lambda}^{(j)})\|_2^2.\tag{2.72}$$

Moreover, note that at any iteration j

$$
\begin{aligned}
L(\boldsymbol{\lambda}^*) \geqslant L(\boldsymbol{\lambda}^{(j)}) &\overset{(a)}{=} \sum_{q\in Q} p^a(q) L_q(\boldsymbol{\lambda}^{(j)}) - \langle \boldsymbol{\lambda}^{(j)}, S\rangle \\
&\overset{(b)}{=} \sum_{q\in Q} p^a(q) \left(\langle \mathbf{c}, \mathbf{x}[q]^{(j)}\rangle + \langle \boldsymbol{\lambda}^{(j)}, g_q(\boldsymbol{\lambda}^{(j)})\rangle\right) - \langle \boldsymbol{\lambda}^{(j)}, S\rangle \\
&= \sum_{q\in Q} p^a(q) \langle \mathbf{c}, \mathbf{x}[q]^{(j)}\rangle + \langle \boldsymbol{\lambda}^{(j)}, G(\boldsymbol{\lambda}^{(j)})\rangle \\
&= \left\langle \mathbf{c}, \sum_{q\in Q} p^a(q)\mathbf{x}[q]^{(j)} \right\rangle + \langle \boldsymbol{\lambda}^{(j)}, G(\boldsymbol{\lambda}^{(j)})\rangle, \qquad (2.73)
\end{aligned}
$$

where (a) holds by the decomposition (2.20) and (b) holds because $\mathbf{x}[q]^{(j)}$ is an optimal solution of $L_q(\boldsymbol{\lambda}^{(j)})$. Thus, we have that at iteration j

$$
\begin{aligned}
L(\boldsymbol{\lambda}^*) - \left\langle \mathbf{c}, \sum_{q\in Q} p^a(q)\mathbf{x}[q]^{(j)} \right\rangle &\overset{(a)}{\geqslant} \langle \boldsymbol{\lambda}^{(j)}, G(\boldsymbol{\lambda}^{(j)})\rangle \\
&\overset{(b)}{\geqslant} \frac{1}{2\alpha_J}\left(\|\boldsymbol{\lambda}^{(j+1)}\|_2^2 - \|\boldsymbol{\lambda}^{(j)}\|_2^2 - \alpha_J^2 \|G(\boldsymbol{\lambda}^{(j)})\|_2^2\right),
\end{aligned}
$$

where (a) holds from (2.73) and (b) follows from (2.72). Therefore, after averaging over all $j = 1, \ldots, J$ iterations and denoting by $\bar{\mathbf{x}} = \frac{1}{J}\sum_{j=1}^{J}\sum_{q\in Q} p^a(q)\mathbf{x}[q]^{(j)}$, we have that

$$
\begin{aligned}
OPT^E - \langle \mathbf{c}, \bar{\mathbf{x}}\rangle &= L(\boldsymbol{\lambda}^*) - \langle \mathbf{c}, \bar{\mathbf{x}}\rangle \\
&\geqslant \frac{1}{2J\alpha_J}\left(\|\boldsymbol{\lambda}^{(J+1)}\|_2^2 - \|\boldsymbol{\lambda}^{(1)}\|_2^2 - \alpha_J^2 \sum_{j=1}^{J} \|G(\boldsymbol{\lambda}^{(j)})\|_2^2\right) \\
&\overset{(a)}{\geqslant} -\frac{C\cdot OPT^E}{2}\alpha_J,
\end{aligned}
$$

where (a) holds because $\boldsymbol{\lambda}^{(1)} = \mathbf{0}$ and by the assumption $\|G(\boldsymbol{\lambda}^{(j)})\|_2^2 \leqslant C \cdot OPT^E$. Rearranging the terms, we have (2.22), i.e.,

$$
\langle \mathbf{c}, \bar{\mathbf{x}}\rangle \leqslant \left(1 + \frac{C}{2}\alpha_J\right) OPT^E.
$$

We now prove (2.23). First, note that by using (2.71), we have that

$$\|\boldsymbol{\lambda}^{(j+1)}\|_2^2 \leqslant \|\boldsymbol{\lambda}^{(j)}\|_2^2 + \alpha_J^2 \|G(\boldsymbol{\lambda}^{(j)})\|_2^2 + 2\alpha_J \langle \boldsymbol{\lambda}^{(j)}, G(\boldsymbol{\lambda}^{(j)}) \rangle, \tag{2.74}$$

and, from (2.73), we have that

$$OPT^E = L(\boldsymbol{\lambda}^*) \geqslant \left\langle \mathbf{c}, \sum_{q \in Q} p^a(q) \mathbf{x}[q]^{(j)} \right\rangle + \langle \boldsymbol{\lambda}^{(j)}, G(\boldsymbol{\lambda}^{(j)}) \rangle,$$

which implies that, at iteration j

$$\langle \boldsymbol{\lambda}^{(j)}, G(\boldsymbol{\lambda}^{(j)}) \rangle \leqslant OPT^E - \left\langle \mathbf{c}, \sum_{q \in Q} p^a(q) \mathbf{x}[q]^{(j)} \right\rangle \leqslant OPT^E.$$

Let $\hat{C} = OPT^E$. Since OPT^E is independent of the number of iterations J, $\langle \boldsymbol{\lambda}^{(j)}, G(\boldsymbol{\lambda}^{(j)}) \rangle \leqslant \hat{C}$. Therefore, from (2.74), we have that at iteration j

$$\|\boldsymbol{\lambda}^{(j+1)}\|_2^2 \leqslant \|\boldsymbol{\lambda}^{(j)}\|_2^2 + \alpha_J^2 (C \cdot OPT^E) + 2\alpha_J \hat{C}. \tag{2.75}$$

Summing the previous inequality over all J iterations (and using the fact that $\boldsymbol{\lambda}^{(1)} = \mathbf{0}$), we obtain

$$\|\boldsymbol{\lambda}^{(J+1)}\|_2^2 \leqslant J\alpha_J^2 (C \cdot OPT^E) + 2J\alpha_J \hat{C}.$$

In other terms, we have that

$$\frac{1}{J^2 \alpha_J^2} \|\boldsymbol{\lambda}^{(J+1)}\|_2^2 \leqslant \frac{1}{J} (C \cdot OPT^E) + \frac{1}{J\alpha_J} 2\hat{C}. \tag{2.76}$$

Now, note that at iteration j, we have

$$\boldsymbol{\lambda}^{(j+1)} = \max\{0, \boldsymbol{\lambda}^{(j)} + \alpha_J G(\boldsymbol{\lambda}^{(j)})\} \geqslant \boldsymbol{\lambda}^{(j)} + \alpha_J G(\boldsymbol{\lambda}^{(j)}),$$

i.e.,

$$\boldsymbol{\lambda}^{(j+1)} - \boldsymbol{\lambda}^{(j)} \geqslant \alpha_J G(\boldsymbol{\lambda}^{(j)}).$$

Averaging the previous inequality over all J iterations (and using the fact that $\boldsymbol{\lambda}^{(1)} = 0$), we obtain

$$\begin{aligned}\frac{\boldsymbol{\lambda}^{(J+1)}}{J} &\geqslant \alpha_J \left(\frac{1}{J} \sum_{j=1}^{J} G(\boldsymbol{\lambda}^{(j)}) \right) \\ &= \alpha_J \left[\left(\frac{1}{J} \sum_{j=1}^{J} \sum_{q \in Q} p^a(q) g_q(\boldsymbol{\lambda}^{(j)}) \right) - \boldsymbol{S} \right]. \end{aligned} \tag{2.77}$$

Moreover, note that for any $i \in I$ and $k \in K$,

$$\begin{aligned}\frac{1}{J} \sum_{j=1}^{J} \sum_{q \in Q} p^a(q) \left(\sum_{m:(i,k) \in m, m \sim q} x_m^{(j)} \right) &= \sum_{m:(i,k) \in m} \left(\frac{1}{J} \sum_{j=1}^{J} \sum_{q \in Q} p^a(q) x_m^{(j)} \mathbb{1}\{m \sim q\} \right) \\ &= \sum_{m:(i,k) \in m} \bar{x}_m,\end{aligned}$$

where the last equality holds by definition of $\bar{\mathbf{x}}$. Therefore, for any $i \in I$ and $k \in K$, we have that

$$\begin{aligned}\sum_{m:(i,k) \in m} \bar{x}_m - S_{ik} &\overset{(a)}{\leqslant} \frac{\lambda_{ik}^{(J+1)}}{J\alpha_J} \\ &\leqslant \frac{\|\boldsymbol{\lambda}^{(J+1)}\|_2}{J\alpha_J} \\ &\overset{(b)}{\leqslant} \sqrt{\frac{1}{J}(C \cdot OPT^E) + \frac{1}{J\alpha_J} 2\hat{C}} \\ &\overset{(c)}{\leqslant} \frac{\bar{C}}{\sqrt{J\alpha_J}},\end{aligned}$$

where (a) holds by (2.77); (b) follows from (2.76); and (c) holds for J sufficiently large and some positive constant $\bar{C}$ independent of J. Finally, given the previous set of inequalities, we can see that whenever $\bar{C} \geqslant \sqrt{OPT^E(2 + C)}$, then, (2.23) holds true. This completes the proof. ■

Proof of Theorem 2.3:

By using Theorem 2.2, we have that for $\alpha_J = J^{-1/3}$

$$\langle \mathbf{c}, \bar{\mathbf{x}} \rangle \leqslant \left(1 + \frac{C}{2J^{1/3}}\right) OPT^E. \tag{2.78}$$

Now, consider the feasible solution $\mathbf{y} = \{y_m\}_{m \in M}$ as defined in (2.26), with τ as in (2.25). Suppose $\tau \neq 1$ (else, $\mathbf{y} = \bar{\mathbf{x}}$). Then, the expected cost of $\mathbf{y}$ can be bounded as follows:

$$\begin{aligned}
\langle \mathbf{c},\ \mathbf{y} \rangle &\overset{(a)}{=} \sum_{m:m\sim q} c_m \bar{x}_m \tau^{-1} + c_{m'(q)} \left(p^a(q) - \sum_{m:m\sim q} \bar{x}_m \tau^{-1} \right) \\
&\overset{(b)}{=} \sum_{m:m\sim q} c_m \bar{x}_m \tau^{-1} + c_{m'(q)} \left(\sum_{m:m\sim q} \bar{x}_m - \sum_{m:m\sim q} \bar{x}_m \tau^{-1} \right) \\
&= \sum_{m:m\sim q} c_m \bar{x}_m \tau^{-1} + \sum_{m:m\sim q} c_{m'(q)} \left(\bar{x}_m - \bar{x}_m \tau^{-1} \right) \\
&\overset{(c)}{\leqslant} \sum_{m:m\sim q} c_m \bar{x}_m \tau^{-1} + \sum_{m:m\sim q} \kappa c_m \left(\bar{x}_m - \bar{x}_m \tau^{-1} \right) \\
&= \sum_{m:m\sim q} c_m \bar{x}_m \cdot \left((1-\kappa)\tau^{-1} + \kappa \right) \\
&\overset{(d)}{\leqslant} \sum_{m:m\sim q} c_m \bar{x}_m \cdot \frac{sJ^{1/3} + \kappa(sJ^{1/3} + \bar{C}) - \kappa sJ^{1/3}}{sJ^{1/3} + \bar{C}} \\
&= \sum_{m:m\sim q} c_m \bar{x}_m \cdot \frac{sJ^{1/3} + \kappa \bar{C}}{sJ^{1/3} + \bar{C}} \\
&= \left(\frac{sJ^{1/3} + \kappa \bar{C}}{sJ^{1/3} + \bar{C}} \right) \langle \mathbf{c}, \bar{\mathbf{x}} \rangle
\end{aligned}$$

where (a) holds by using definition (2.26) and the fact that $\sum_{m:m\sim q} \bar{x}_m = p^a(q)$; (b) holds by using (2.24); (c) follows from the assumption that for some $\kappa > 1$, $c_{m'(q)}/c_m \leqslant \kappa$, for all $m \sim q$ and all $q \in Q$; and (d) holds from the fact that $(1-k) < 0$ and because, from (2.23), we have that $\tau \leqslant 1 + \bar{C}/(sJ^{1/3})$, with $s = \min_{i\in I, k\in K, S_{ik}>0}\{S_{ik}\}$.

Therefore, from (2.78), we have that

$$\langle \mathbf{c},\ \mathbf{y} \rangle \leqslant \left(\frac{sJ^{1/3} + \kappa \bar{C}}{sJ^{1/3} + \bar{C}} \right) \langle \mathbf{c}, \bar{\mathbf{x}} \rangle \leqslant \left(\frac{sJ^{1/3} + \kappa \bar{C}}{sJ^{1/3} + \bar{C}} \right) \left(1 + \frac{C}{2J^{1/3}} \right) OPT^E.$$

Now, following similar steps to the proof of Lemma 2.3 and by Proposition 2.1, we have that when Algorithm 1 uses Algorithm 2 as a subroutine in Step 2 of the online process achieves a competitive ratio of at most

$$\left(\frac{sJ^{1/3} + \kappa \bar{C}}{sJ^{1/3} + \bar{C}} \right) \left(1 + \frac{C}{2J^{1/3}} \right) \left(1 + \frac{(\kappa - 1)|q_{\max}|}{\sqrt{s+3}} \right).$$

This completes the proof. ■

Proof of Proposition 2.4:

We start by proving that for any given γ-conservative method-acceptance strategy, Algorithm 4 achieves a competitive ratio of at least γ. To prove this, note that at time t, the expected reward of Algorithm 4 is given by

$$\sum_{q \in Q} r_q \mathbb{P}(\text{accept order } q \text{ at time } t \mid \mathcal{D}^{qt} = 1) p_t(q).$$

Moreover

$$\begin{aligned} &\mathbb{P}(\text{accept order } q \text{ at time } t \mid \mathcal{D}^{qt} = 1) \\ &= \mathbb{P}(\text{accept order } q \text{ at time } t \mid Y_q^t = 1, \mathcal{D}^{qt} = 1) \mathbb{P}(Y_q^t = 1 \mid \mathcal{D}^{qt} = 1) \\ &\overset{(a)}{\geqslant} \gamma \frac{y_q^t}{p_t(q)}, \end{aligned}$$

where (a) holds by definition of γ-conservative method-acceptance strategy (adapted to the NRM setting, where we have a single method for each order type q). Therefore, using $\mathbb{E}\left[\mathrm{ALG}_4(\mathcal{D})\right]$ to denote the expected reward collected from Algorithm 4, we

have

$$\frac{\mathbb{E}\left[\mathrm{ALG}_4(\mathcal{D})\right]}{\mathbb{E}\left[\mathrm{OPT}(\mathcal{D})\right]} \overset{(a)}{\geqslant} \frac{\mathbb{E}\left[\mathrm{ALG}_4(\mathcal{D})\right]}{\sum_{t\in[T]}\sum_{q\in Q} r_q y_q^t}$$

$$\geqslant \frac{\sum_{t\in[T]}\sum_{q\in Q} r_q \left(\gamma \frac{y_q^t}{p_t(q)}\right) p_t(q)}{\sum_{t\in[T]}\sum_{q\in Q} r_q y_q^t} = \gamma,$$

where (a) holds because $\sum_{t\in[T]}\sum_{q\in Q} r_q y_q^t$ is an upper bound on $\mathbb{E}\left[\mathrm{OPT}(\mathcal{D})\right]$. Now, combining this result with the γ-conservative method-acceptance strategy provided in Proposition 2.1 (appropriately adapted to the special case in which we have a single facility), completes the proof. ■

3

Multi-Item Order Fulfillment with Reusable Resources

In this chapter, we expand our multi-item fulfillment model from the previous chapter to incorporate scenarios where inventory is reusable. In the literature, these models are referred to as models with "reusable resources". This framework finds relevance in various applications such as sports equipment rentals (like scuba diving, skiing/snowboarding, or climbing gear), photography equipment rentals, and comprehensive party and event supplies rentals (like rentals of tables, chairs, linens, tableware, decorations, and audio-visual equipment). Reusable resources models have been studied in several settings, including revenue management problems, e.g., Levi and Radovanović (2010), and assortment problems, e.g., Feng et al. (2022).

3.1 Offline Formulation and Algorithm

In this section, we consider the model we presented in the previous chapter (see Section 2.2 for details), with the only difference that now inventory can be reused. In order to capture the dynamics of reusable inventory, we define $G_{ik} : \mathbb{R}_+ \to [0, 1]$ to be the cumulative distribution function (CDF) of rental duration for item i from

facility k. For simplicity of exposition, we assume that rental durations are stationary across time and independent of each other. Moreover, we let $R_{ik}(t,\tau)$ be the binary random variable which is 1 if the rental duration for pair (i,k) at time τ is at least $t-\tau$, with $\tau \leqslant t$ and $t \in [T]$. Clearly, $\mathbb{E}[R_{ik}(t,\tau)] = 1 - G_{ik}(t-\tau)$. Given these definitions, the LP relaxation of the offline problem in the reusable inventory setting can be written as follows:

$$\min_{\mathbf{x}} \sum_{t\in[T]} \sum_{q\in Q} \sum_{m:m\sim q} c_m x_m^t \tag{3.1}$$

$$\text{s.t.} \sum_{m:m\sim q} x_m^t = p_t(q),\ \forall q \in Q, t \in [T], \tag{3.2}$$

$$\sum_{\tau\in[t]} \sum_{m:(i,k)\in m} (1 - G_{ik}(t-\tau)) x_m^\tau \leqslant S_{ik},\ \forall i \in I, k \in K, t \in [T], \tag{3.3}$$

$$x_m^t \in [0,1],\ \forall m \in M, t \in [T]. \tag{3.4}$$

Note that this LP is similar to the LP relaxation (2.6) in the non-reusable setting, except for the inventory constraint. Indeed, the set of constraints (3.3) ensures that, in expectation, **at any time** t, the amount of inventory in use does not exceed its capacity, while the set of constraints (2.8) in the non-reusable setting only ensures that, in expectation, **over the entire time horizon** $[T]$, we do not fulfill orders using more than S_{ik} units of item $i \in I$ from facility $k \in K$. Essentially, the difference lies in the fact that in the non-reusable setting, ensuring that we do not run out of inventory by the end of the time horizon is enough to ensure that we do not run out of inventory at any time t. However, this is not anymore the case in the reusable resources setting because now resources can be used for a random period of time. Note also that in scenarios where rental durations are indefinitely long, i.e., infinite, our model simplifies back to the original non-reusable case (LP relaxation (2.6)), as both models share the same feasible set and objective.

In this reusable resources setting, we now show how to find a γ-conservative

method-acceptance strategy for step 2 of the online process of Algorithm 1 (after having solved the LP relaxation (3.1) in the offline process of Algorithm 1). For details about γ-conservative method-acceptance strategies, please refer to Section 2.4. The main idea behind the γ-conservative method-acceptance strategy we adopt in the reusable inventory setting is as follows. Suppose that m is the sampled method at time t in the online process of Algorithm 1. Then, for each $(i,k) \in m$, an independent procedure is used to decide whether to make pair $(i,k) \in m$ available for fulfillment or not, given the history up to time t. If all $(i,k) \in m$ are made available and are in-stock, then we accept method m for fulfillment. Otherwise, we fulfill the order using the dummy facility. Next, we present this method-acceptance strategy and formally prove that it is indeed a γ-conservative method-acceptance strategy, as defined in (2.3).

Algorithm 7 A method-acceptance strategy for reusable inventory

For $t = 1, \ldots, T$:

- Let m be the method that is drawn at time t.
- For each $(i,k) \in m$, toss an independent coin and make (i,k) available for fulfillment with probability γ_{ik}.
- If *all* $(i,k) \in m$ are available for fulfillment and in-stock, then accept method m. Else, reject m.

Proposition 3.1. *When* $\gamma_{ik} = 1 - \sqrt{\log(s)/s}$ *for all* (i,k)*, Algorithm 7 provides a* γ*-conservative method-acceptance strategy for step* 2 *of the online process of Algorithm 1 with*

$$\gamma = 1 - 2|q_{\max}|\sqrt{\log(s)/s},$$

where $|q_{\max}|$ *denotes the size of the largest possible order and* $s = \min_{i\in I, k\in K, S_{ik}>0}\{S_{ik}\} \geqslant 3$.

3.2 Proof of Proposition 3.1

In this section, we present a detailed proof of Proposition 3.1.

Proof: We want to prove that Algorithm 7 is a γ-conservative method-acceptance strategy, i.e.,

$$\mathbb{P}(\text{accepting the method at } t \mid \text{the method is of type } m) \geqslant \gamma,$$

with $\gamma = 1 - 2|q_{\max}|\sqrt{\log(s)/s}$. We start by proving that for all (i,k) and $t \in [T]$

$$\mathbb{P}\left((i,k) \text{ is out-of-stock at time } t\right) \leqslant \exp\left(-\frac{(1-\gamma_{ik})^2 s}{1+\gamma_{ik}}\right). \tag{3.5}$$

In order to do so, we define the following events:

- I_{ik}^{t} : event that one unit of (i,k) is allocated for fulfillment at time t;
- $\mathrm{I}_{ik}^{t(1)}$: event that (i,k) is in the sampled method m (in Algorithm 7) at time t;
- $\mathrm{I}_{ik}^{t(2)}$: event that (i,k) is available for fulfillment at time t;
- $\mathrm{I}_{ik}^{t(3)}$: event that one unit of (i,k) is available in the inventory at the beginning of time t;
- $\mathrm{I}_{ik}^{\tau t}$: event that the rental duration of (i,k) at time τ is at least $t-\tau$.

Note that, by definition, we have

$$\mathrm{I}_{ik}^{t} = \mathrm{I}_{ik}^{t(1)}\mathrm{I}_{ik}^{t(2)}\mathrm{I}_{ik}^{t(3)},$$

and the event that (i,k) is out-of-stock at time t can be rewritten as

$$\sum_{\tau=1}^{t-1} \mathrm{I}_{ik}^{\tau}\mathrm{I}_{ik}^{\tau t} \geqslant S_{ik}.$$

Given this notation, we want to prove

$$\mathbb{P}\left(\sum_{\tau=1}^{t-1} \mathrm{I}_{ik}^{\tau}\mathrm{I}_{ik}^{\tau t} \geqslant S_{ik}\right) \leqslant \exp\left(-\frac{(1-\gamma_{ik})^2 s}{1+\gamma_{ik}}\right).$$

However, note that $\{\mathrm{I}_{ik}^{t}\}_t$ are not independent across t and I_{ik}^{τ} and $\mathrm{I}_{ik}^{\tau t}$ are not independent at time τ. Because of this dependent structure, consider a hypothetical scenario where we run Algorithm 7 *without* inventory constraints, and define random variables $\{\bar{\mathrm{I}}_{ik}^{t}, \bar{\mathrm{I}}_{ik}^{t(1)}, \bar{\mathrm{I}}_{ik}^{t(2)}, \bar{\mathrm{I}}_{ik}^{t(3)}, \bar{\mathrm{I}}_{ik}^{\tau t}\}$ as above. Specifically, $\bar{\mathrm{I}}_{ik}^{t(1)} = \mathrm{I}_{ik}^{t(1)}, \bar{\mathrm{I}}_{ik}^{t(2)} = \mathrm{I}_{ik}^{t(2)}$ and $\bar{\mathrm{I}}_{ik}^{\tau t} = \mathrm{I}_{ik}^{\tau t}$. Now, by definition, we have that

$$\bar{\mathrm{I}}_{ik}^{t} = \bar{\mathrm{I}}_{ik}^{t(1)}\bar{\mathrm{I}}_{ik}^{t(2)},$$

since $\bar{\mathrm{I}}_{ik}^{t(3)} = 1$ for all t, and $\{\bar{\mathrm{I}}_{ik}^{\tau}\bar{\mathrm{I}}_{ik}^{\tau t}\}_\tau$ are independent across τ. Moreover, because of this coupling, we have that

$$\sum_{\tau=1}^{t-1} \mathrm{I}_{ik}^{\tau}\mathrm{I}_{ik}^{\tau t} \;\leqslant\; \sum_{\tau=1}^{t-1} \bar{\mathrm{I}}_{ik}^{\tau}\bar{\mathrm{I}}_{ik}^{\tau t},$$

for all (i,k) and $t \in [T]$. Thus, since $\mathbb{P}\left(\sum_{\tau=1}^{t-1} \mathrm{I}_{ik}^{\tau}\mathrm{I}_{ik}^{\tau t} \geqslant S_{ik}\right) \leqslant \mathbb{P}\left(\sum_{\tau=1}^{t-1} \bar{\mathrm{I}}_{ik}^{\tau}\bar{\mathrm{I}}_{ik}^{\tau t} \geqslant S_{ik}\right)$, it is enough to prove that for all (i,k) and $t \in [T]$

$$\mathbb{P}\left(\sum_{\tau=1}^{t-1} \bar{\mathrm{I}}_{ik}^{\tau}\bar{\mathrm{I}}_{ik}^{\tau t} \geqslant S_{ik}\right) \leqslant \exp\left(-\frac{(1-\gamma_{ik})^2 s}{1+\gamma_{ik}}\right).$$

Now, note that

$$\begin{aligned}
&\mathbb{E}[\bar{\mathrm{I}}_{ik}^{\tau}\bar{\mathrm{I}}_{ik}^{\tau t}] \\
&= \mathbb{E}[\bar{\mathrm{I}}_{ik}^{\tau}]\mathbb{E}[\bar{\mathrm{I}}_{ik}^{\tau t}] \\
&= \mathbb{E}[\bar{\mathrm{I}}_{ik}^{\tau(1)}\bar{\mathrm{I}}_{ik}^{\tau(2)}]\mathbb{E}[\bar{\mathrm{I}}_{ik}^{\tau t}] \\
&= \mathbb{E}[\mathbb{E}[\bar{\mathrm{I}}_{ik}^{\tau(2)} \mid \bar{\mathrm{I}}_{ik}^{\tau(1)}]]\mathbb{E}[\bar{\mathrm{I}}_{ik}^{\tau t}] \\
&= \gamma_{ik}\mathbb{E}[\bar{\mathrm{I}}_{ik}^{\tau(1)}]\mathbb{E}[\bar{\mathrm{I}}_{ik}^{\tau t}] \\
&= \gamma_{ik}\sum_{m:(i,k)\in m} y_m^{\tau}(1 - G_{ik}(t-\tau)).
\end{aligned}$$

Thus,

$$\mathbb{E}\left[\sum_{\tau=1}^{t-1}\bar{\mathrm{I}}_{ik}^{\tau}\bar{\mathrm{I}}_{ik}^{\tau t}\right] = \gamma_{ik}\sum_{\tau=1}^{t-1}\sum_{m:(i,k)\in m} y_m^{\tau}(1 - G_{ik}(t-\tau)) \leqslant \gamma_{ik}S_{ik}.$$

Therefore, applying the Multiplicative Chernoff Bound to $\{\bar{\mathrm{I}}_{ik}^{\tau}\bar{\mathrm{I}}_{ik}^{\tau t}\}_{\tau=1}^{t-1}$, one can check that we obtain

$$\mathbb{P}\left(\sum_{\tau=1}^{t-1}\bar{\mathrm{I}}_{ik}^{\tau}\bar{\mathrm{I}}_{ik}^{\tau t} \geqslant S_{ik}\right) \leqslant \exp\left(-\frac{(1-\gamma_{ik})^2 s}{1+\gamma_{ik}}\right),$$

for all (i,k) and $t \in [T]$.

Now, consider time t and pair (i,k). Denoting by

$$A_{ik}^t = \{\text{inventory } (i,k) \text{ is available for fulfillment at } t\},$$

the event in which (i,k) is available for fulfillment at time t, and by

$$B_{ik}^t = \{(i,k) \text{ is in-stock at } t\},$$

the event in which one unit of (i,k) is available in the inventory at time t, we have that, according to Algorithm 7, for any $m \in M$ and $t \in [T]$:

$$
\begin{aligned}
&\mathbb{P}(\text{accepting the method at } t \mid \text{the method is of type } m) \\
&\overset{(a)}{=} \mathbb{P}\left(\bigcap_{(i,k)\in m} \{A_{ik}^t \cap B_{ik}^t\}\right) \\
&\overset{(b)}{\geqslant} 1 - \sum_{(i,k)\in m} (1 - \mathbb{P}(A_{ik}^t)\mathbb{P}(B_{ik}^t)) \\
&\overset{(c)}{\geqslant} 1 - \sum_{(i,k)\in m} \left(1 - \gamma_{ik}\left(1 - \exp\left(-\frac{(1-\gamma_{ik})^2 s}{1+\gamma_{ik}}\right)\right)\right) \\
&\overset{(d)}{\geqslant} 1 - \sum_{(i,k)\in m} 2\sqrt{\log(s)/s} \\
&\geqslant 1 - 2|q_{\max}|\sqrt{\log(s)/s},
\end{aligned}
$$

where (a) follows from the definition of Algorithm 7; (b) holds by using union bound and independence; (c) follows from the definition of Algorithm 7 and from (3.5); and (d) holds because, if we choose $\gamma_{ik}^* = 1 - \sqrt{\log(s)/s}$, one can check that

$$
\begin{aligned}
1 - \gamma_{ik}\left(1 - \exp\left(-\frac{(1-\gamma_{ik})^2 s}{1+\gamma_{ik}}\right)\right) &= \sqrt{\log(s)/s} + s^{\frac{1}{-2+\sqrt{\log(s)/s}}} - s^{\frac{1}{-2+\sqrt{\log(s)/s}}}\sqrt{\log(s)/s} \\
&\leqslant \sqrt{\log(s)/s} + s^{-1/2} \\
&\leqslant 2\sqrt{\log(s)/s},
\end{aligned}
$$

where the last inequality holds if $s \geqslant 3$. This completes the proof. ■

We should highlight that the above proposition holds true for $s = 2$ as well by letting $\gamma = 1 - 2.02625|q_{\max}|\sqrt{\log(s)/s}$ (instead of $1 - 2|q_{\max}|\sqrt{\log(s)/s}$). For simplicity, we stated the theorem with $s \geqslant 3$.

3.3 Main Result

In this section, we formally prove the performance guarantee in the reusable inventory setting when Algorithm 1 uses Algorithm 7 as a subroutine in Step 2 of the online

process. The proof is a simple application of Lemma 2.2.

Theorem 3.1. *For any online multi-item fulfillment problem with reusable inventory, when Algorithm 1 solves* (3.1) *in the offline process and uses Algorithm 7 as a subroutine in Step 2 of the online process, we obtain a fulfillment strategy with a competitive ratio of at most*

$$1 + 2(\kappa - 1)|q_{\max}|\sqrt{\frac{\log(s)}{s}}, \tag{3.6}$$

where $|q_{\max}|$ *denotes the size of the largest possible order and* $s = \min_{i\in I, k\in K, S_{ik}>0}\{S_{ik}\} \geqslant 3$.

Proof of Theorem 3.1: Let $\mathbb{E}\left[\mathrm{ALG}_1(\mathcal{D})\right]$ denote the expected cost incurred by Algorithm 1. Then, using the γ-conservative method-acceptance strategy defined in Algorithm 7 with $\gamma = 1 - 2|q_{\max}|\sqrt{\log(s)/s}$ (as proved in Proposition 3.1), we have that under Algorithm 1

$$\frac{\mathbb{E}\left[\mathrm{ALG}_1(\mathcal{D})\right]}{\mathbb{E}\left[\mathrm{OPT}(\mathcal{D})\right]} \overset{(a)}{\leqslant} 1 + (\kappa - 1)(1 - \gamma) = 1 + 2(\kappa - 1)|q_{\max}|\sqrt{\frac{\log(s)}{s}},$$

where (a) follows from Lemma 2.2. This completes the proof. ■

Finally, note that, an ϵ-approximation of problem (3.1) is enough for the above strategy to still achieve approximately the same competitive ratio (i.e., one can first obtain an ϵ-approximation of (3.1) and apply Lemma 2.3 instead of Lemma 2.2). Moreover, we can also easily relax $|q_{\max}|$ to $\mathbb{E}_{q\sim F(\cdot)}[|q|]$, similarly to how we did it in Section 2.12.1.

4

Online Recommendations: A Contextual Bandit Approach

4.1 Introduction

In this chapter, we shift our focus to another critical element of decision-making in online marketplaces: recommendations. As digital marketplaces evolve, the importance of recommendation systems becomes increasingly apparent. With online consumer purchases now accounting for 16% of the market and e-commerce valuations exceeding $200 billion (e.g., Amazon alone has over 110 million Prime customers[1]), the impact of advanced recommendation algorithms on user experience and sales performance is undeniable. Recent advancements in Artificial Intelligence and Machine Learning have particularly empowered digital platforms to harness consumer information for providing personalized recommendations. Examples include product recommendations on e-commerce platforms such as Amazon, movie recommendations on streaming platforms such as Netflix, or music recommendations on music streaming platforms such as Spotify.

[1] US Census Bureau News, U.S. Department of Commerce.

Companies such as Amazon, Google, Meta or Netflix provide an extensive array of choices to their users in terms of products and services. However, navigating through this vast sea of options can become overwhelming for consumers. Therefore, to enhance user experience, it is essential for these entities to develop effective personalized recommendation systems that can accurately match users with options that align with their preferences. This task, however, is challenging for multiple reasons. Firstly, companies cannot directly observe user preferences. Secondly, quantifying a user's willingness to pay or their satisfaction from a specific product or service is complex. Moreover, not only the valuation of each customer is unknown, but also the decision-making process with which each user determines their value of the product is unknown. Consequently, decision makers are faced with the problem of "competing" with a large class of mappings (which is potentially unknown) from the users' characteristics (also known as side information) to their actual value for the product. Accurately estimating these customer valuations is crucial, as they significantly influence the online user experience and, by extension, the pricing strategies and revenues of these platforms. This chapter specifically tackles the following question: how should an online platform decide, based on information from past user interactions, which products to offer to new customers as they arrive?

In this chapter, we consider a decision maker recommending a product to costumers over a finite time horizon. In each period, a new customer arrives, and the decision maker must decide which characteristics of the product to show. Note that we focus our attention to a single product (e.g., movies) and try to design an algorithm that recommends features for that product. However, our approach can be easily extended to multiple products. In our model, the decision maker can base their recommendation decision based on the data about the customer (called hereafter context or side information), and the history of past recommendations. Once a recommendation is selected, the corresponding valuation of the customer is realized.

Since the decision maker does not know the valuation of the arriving customer, we do not impose any constraints on how the side information determines the market value of the product. Therefore, in this chapter, the decision maker is competing with a large class of possible mappings from side information to the actual market values.

Given the above framework, we consider an online regret minimization formulation of this problem in which the decision maker competes with the best possible policy that models the set of functions mapping side information to market valuations. We find the regret bound of our online policy against an arbitrary class of policies in terms of the Rademacher complexity of the class in question. Our bound exhibits the so-called approximation versus statistical error trade-off: a larger class of policies is closer to the actual best mapping in practice, but it is harder for the algorithm to compete against it. We do not assume any constraints on the set of mappings and instead develop our algorithm and regret analysis such that it can adapt to any specific class of policies. We then apply this framework to two specific classes of policies: (i) class of finite mappings from contexts to actions and (ii) the class of policies characterized by linear mappings from contexts to values. In the full-feedback setting, our regret scales as $O(\sqrt{n})$ (and we show it is tight), where n is the length of time horizon and depends on the number of mappings in case (i) and the sparsity of the linear mapping in case (ii). In the bandit setting, our regret scales as $O(n^{2/3})$ and again depends on the number of mappings in case (i) and the sparsity of the linear mappings in case (ii).

4.1.1 Related Literature

Our work relates to the online learning literature (see, for example, Cesa-Bianchi and Lugosi (2006)), and, in particular, since we formulate our problem as a regret minimization model with side information, it relates to the contextual bandit literature. A contextual bandit problem is an online learning problem where decisions are made

with the aid of side information, and actions are guided by a set of policies. This perspective is useful because one can apply classical algorithms, such as Hedge (see, e.g., Freund and Schapire (1997) and Cesa-Bianchi et al. (1997)) and Exp4 (see, e.g., Auer et al. (1995)), which give information theoretically optimal regret bounds of $O(\sqrt{n \log(|\mathcal{F}|)})$ in full-information feedback and $O(\sqrt{nK \log(|\mathcal{F}|)})$ in the bandit feedback setting, where n is the total number of rounds, K is the total number of actions, and $\mathcal{F}$ is the policy set. However, directly applying standard online learning algorithms to the contextual setting leads to a running time that is linear in the number of policies. Given that the optimal regret is only logarithmic in $|\mathcal{F}|$ and that our goal is to learn a very rich policy class, we want to consider policy classes that are exponentially large. When we use a large policy class, existing algorithms are no longer computationally tractable. To overcome this computational challenge several papers such as Langford and Zhang (2007), Dudik et al. (2011), Rakhlin et al. (2012), Perchet et al. (2013), Agarwal et al. (2014), and Rakhlin and Sridharan (2015) have developed "oracle-based" algorithms that only access the policy class through an optimization oracle for the offline full-information problem. Optimization oracles have been used in designing contextual bandit algorithms (see, e.g., Agarwal et al. (2014), Langford and Zhang (2007), Dudik et al. (2011)) that achieve the optimal $O(\sqrt{nK \log(|\mathcal{F}|)})$ regret while also being computationally efficient (i.e., requiring $\text{poly}(K, \log(|\mathcal{F}|), n)$ oracle calls and computation). However, these results only apply when the values are drawn identically and independently at random at each round, diverging from the computationally inefficient approaches that can handle adversarial inputs. In contrast, in this chapter, we develop an oracle-based algorithm for online recommendations with stochastic side information that can handle adversarial inputs and also works for any class of policies (and not only a finite class of policies).

The papers of Rakhlin and Sridharan (2016) and Syrgkanis et al. (2016) are the works that more closely relate to ours. In these papers, the authors develop a

computationally efficient algorithm based on a relaxation framework with partial information. We use this framework to develop algorithms for two settings: (i) full-feedback and (ii) bandit feedback. The regret bound of our analysis is in terms of the Rademacher complexity of the class of policies in question. We then develop a novel analysis of this Rademacher complexity that enables us to provide explicit characterizations for two special classes of policies, namely the class of finite mappings and the class of policies characterized by linear mappings from contexts to values.

Finally, our work also relates to the emerging literature at the intersection of online learning and operations research. In particular, Cohen et al. (2020) and Lobel et al. (2018) study dynamic pricing with side information when the users' values are liner functions of the contexts; Keskin and Birge (2019) and Keskin and Zeevi (2017) also study dynamic pricing; Niazadeh et al. (2021) and Balseiro et al. (2019) study contextual learning with cross-learning; Bastani et al. (2020) and Bastani and Bayati (2019) study online learning with high-dimensional contexts; Cassel et al. (2018) study risk criteria in online learning; and Foster et al. (2020) study instant-dependent contextual bandits. We depart from this literature by not assuming any specific relationship between the contexts and the users' values, and, instead, we develop a framework that captures the trade-off between approximation and statistical error.

4.1.2 Outline

The rest of the chapter is organized as follows. Section 4.2 describes the model and the notation used throughout the chapter. In Section 4.3, we first present the full-feedback setting in which the customer's valuation is fully revealed to the decision maker and then provide an algorithm and the regret bound for an arbitrary set of functions $\mathcal{F}$, along with specific bounds for two particular choices of $\mathcal{F}$. In Section 4.4, we consider the bandit feedback setting in which the decision maker observes only the customer's valuation for the recommended features and then provide an

algorithm and the regret bound for an arbitrary set of functions $\mathcal{F}$. Similarly to the full-feedback setting, we provide bounds for two particular choices of $\mathcal{F}$. Finally, Section 4.6 presents the proofs of the statements from the text.

4.2 Model

Consider a setting in which customers (throughout we use the terms customer and user interchangeably) arrive sequentially to a platform which offers a product (or service) over a finite time horizon n. At each time period $t = 1, \ldots, n$, a customer arrives and the platform (throughout we use the terms platform and decision maker interchangeably) recommends a product with a certain set of features. The customer that arrives at time t is represented by a vector of side information $x_t \in \mathcal{X} \subseteq \mathbb{R}^N$ and has a private market value $v_t = v(x_t) \in [-1, 1]^d$ for each of the d features of the product. The side information x_t captures the characteristics (or data) of the customer such as location, gender, age, and purchase history. The market value v_t is the valuation of the customer for each of the product features (which is a function of their characteristics). Note that if the decision maker knows the market value, the optimal recommendation would be to include the features whose values are positive and exclude the ones whose values are negative. The decision-maker, however, does not know the mapping from contexts to values (i.e., does not know v) and, instead, tries to minimize the regret against the optimal mapping in a class of policies, which we will describe next.

The context of the customer arriving at time t is drawn i.i.d. from some known distribution $\mathbb{P}_x$ on $\mathcal{X}$. The market value v_t is chosen adversarially and it is unknown to the decision maker. Specifically, $v : \mathcal{X} \to [-1, 1]^d$ is unknown and it describes the complex mechanism by which a customer assigns a market value to a given product. After a recommendation to user x_t, the platform observes the result of its interaction with the user. In this chapter, we consider two complementary settings: first, we

study the full-feedback case in which the valuation of user x_t is observed at the end of period t, i.e., v_t is fully revealed to the decision maker after the recommendation; second, we study the more realistic bandit feedback case in which, after time t, the decision maker only observes the customer's valuation for the recommended product.

Upon arrival of customer x_t, the platform observes the context x_t and then chooses an action, i.e. a recommendation, $a_t \in \mathcal{A}$. For simplicity, we assume that the platform chooses whether to include some particular feature of the product or not, i.e., the action set is assumed to be $\mathcal{A} \equiv \{0,1\}^d$. Once a recommendation a_t is selected, the corresponding customer valuation v_t is realized. Note that, given this framework, each action $a_t \in \mathcal{A}$ describes a different product that the decision maker can recommend to the arriving customer. Given an action a_t, the valuation of product a_t for customer x_t is given by $a_t^T v_t$. In the full feedback case, the platform observes v_t, while in the bandit feedback case, the platform only observes $a_t^T v_t$. Since we allow for randomized recommendations as well, we denote by q_t the distribution over the 2^d possible recommendations in round t, and draw $a_t \sim q_t$. Given this notation, the expected valuation of product a_t for customer x_t is given by

$$\mathbb{E}_{a_t \sim q_t}\left[a_t^T v_t\right] = q_t^T A v_t, \tag{4.1}$$

where A is a $2^d \times d$ matrix whose rows are all the possible (ordered) actions in $\mathcal{A}$. The goal of the platform is to design a prediction algorithm with large expected cumulative valuation $\sum_{t=1}^{n} q_t^T A v_t$ over the entire time horizon n.

A natural way to encode the decision maker's prior knowledge from past recommendations is to consider a class of functions $\mathcal{F} \coloneqq \{f \mid f : \mathcal{X} \to \mathcal{A}\}$, with the hope that one of the functions in $\mathcal{F}$ will incur large cumulative valuation on the presented contexts, i.e., large $\sum_{t=1}^{n} f(x_t)^T v_t$. Then, the platform's goal is to make predictions

in order to minimize the regret defined as:

$$\textbf{Regret} = \sup_{f \in \mathcal{F}} \sum_{t=1}^{n} f(x_t)^T v_t - \sum_{t=1}^{n} q_t^T A v_t. \tag{4.2}$$

Note that if the modeling choice of $\mathcal{F}$ is appropriate and (4.2) is small, the recommendation algorithm is guaranteed to incur large expected cumulative valuation $\sum_{t=1}^{n} q_t^T A v_t$.

Let us now define some notation. Throughout the chapter, we denote by $(x, y, z)_{1:n}$ a sequence of tuples $\{(x_1, y_1, z_1), \ldots, (x_n, y_n, z_n)\}$; we let Δ_S denote the set of distributions over a set S; and $[n]$ denote the set $\{1, 2, \ldots, n\}$. The vector of ones is denoted by $\mathbf{1}$ and the vector of zeros is denoted by $\mathbf{0}$. The indicator of a set S is denoted by $\mathbb{1}\{S\}$. Another important notation that we use in the bandit feedback setting is $e_{f(x)}$. This (abuse of) notation represents the vector whose components are all zero except for the index corresponding to the vector $f(x) \in \mathcal{A}$.

4.3 Full-Feedback

As a preliminary study, we consider the case in which the decision maker, at each time t, observes the entire vector $v_t \in [-1, 1]^d$, i.e., at the end of round t, v_t is fully revealed to the platform. Specifically, at each round $t \in \{1, \ldots, n\}$, the platform observes side information $x_t \in \mathcal{X}$ drawn from $\mathbb{P}_x$ and chooses a distribution $q_t \in \Delta_{\mathcal{A}}$. The adversary then chooses $v_t \in [-1, 1]^d$ and, finally, the decision maker draws an action $\hat{a}_t \sim q_t$ and observes v_t.

We describe the full-information obtained at round t as a tuple

$$I_t \coloneqq I_t(x_t, q_t, a_t, v_t) = (x_t, q_t, a_t, v_t),$$

where x_t is observed before q_t is chosen and v_t is revealed. Define a full-information relaxation $\mathbf{Rel}^{\mathrm{FF}}(\cdot)$ to be a function that maps $(I_1, \ldots, I_t)$ to a real number, for any

$t \in \{1, \ldots, n\}$. We say that a full-information relaxation $\mathbf{Rel}^{\mathrm{FF}}(\cdot)$ is **admissible** if for any $t \in \{1, \ldots, n\}$ and any $(I_1, \ldots, I_t)$,

$$\mathbb{E}_{x_t} \sup_{q_t} \min_{v_t} \mathbb{E}_{a_t \sim q_t} \left\{ a_t^T v_t + \mathbf{Rel}^{\mathrm{FF}}(I_{1:t-1}, I_t(x_t, q_t, a_t, v_t)) \right\} \geqslant \mathbf{Rel}^{\mathrm{FF}}(I_{1:t-1}), \tag{4.3}$$

and for all $x_{1:n}$, and $v_{1:n}$,

$$\mathbf{Rel}^{\mathrm{FF}}(I_{1:n}) \leqslant -\sup_{f \in \mathcal{F}} \sum_{t=1}^{n} f(x_t)^T v_t. \tag{4.4}$$

Any randomized strategy $(q_t)_{t=1}^n$ that satisfies (4.3) and (4.4) is called an **admissible strategy**. Given this definition, we can prove the following lemma.

Lemma 4.1. *Let* $\mathbf{Rel}^{\mathrm{FF}}(\cdot)$ *be an admissible full-information relaxation and* $(q_t)_{t=1}^n$ *be an admissible strategy. Then, for any* $v_{1:n}$

$$\mathbb{E}_{x_{1:n}}(\mathbf{Regret}) \leqslant -\mathbf{Rel}^{\mathrm{FF}}(\varnothing).$$

Lemma (4.1) provides an upper-bound on the expected regret by means of an admissible full-information relaxation. The difficulty, however, is finding an appropriate $\mathbf{Rel}^{\mathrm{FF}}(\cdot)$ and an admissible strategy $(q_t)_{t=1}^n$. We next present a theorem that provides a way of finding an admissible full-information relaxation.

For any $t \in \{1, \ldots, n\}$, define a $d \times n$ matrix $Y^{(t)}$ as

$$Y^{(t)} := [v_1, \ldots, v_t, 2\epsilon_{t+1}, \ldots, 2\epsilon_n],$$

where each ϵ_s is a d-dimensional vector of independent Rademacher random variables, i.e., each coordinate is an independent Rademacher random variable which is -1 or $+1$ with equal probability. For each $s = 1, \ldots, n$, we denote by $Y_s^{(t)}$ the s-th column of the matrix $Y^{(t)}$. Then, we can prove the following result.

Theorem 4.1. *The full-information relaxation defined as*

$$\mathbf{Rel}^{\mathrm{FF}}(I_{1:t}) \coloneqq \mathbb{E}_{(x,\epsilon)_{(t+1):n}} \inf_{f\in\mathcal{F}} \left\{ -\sum_{s=1}^{n} f(x_s)^T Y_s^{(t)} \right\} \tag{4.5}$$

is admissible. An admissible randomized strategy $(q_t)_{t=1}^n$ *for this relaxation is given in Algorithm 8. The expected regret of the algorithm is upper bounded by*

$$\mathbb{E}_{x_{1:n}} 2\mathscr{R}(\mathcal{F}, x_{1:n}),$$

where $\mathscr{R}(\mathcal{F}, x_{1:n})$ *is defined as*

$$\mathscr{R}(\mathcal{F}, x_{1:n}) \coloneqq \mathbb{E}_{\epsilon_{1:n}} \sup_{f\in\mathcal{F}} \left[\sum_{s=1}^{n} f(x_s)^T \epsilon_s \right]. \tag{4.6}$$

Theorem 4.1 establishes a bound on the regret of the relaxation-based Algorithm 8 in terms of the Rademacher averages for vector-valued functions $\mathscr{R}(\mathcal{F}, x_{1:n})$.

4.3.1 Online Algorithm for Full-Feedback

In this section, we describe an algorithm whose regret is the bound provided in Theorem 4.1. We then discuss the computations involved in this algorithm in Section 4.3.3.

Algorithm 8

- **For** t from 1 to n **do**

  1. Observe x_t, draw $x_{t+1:n} \sim \mathbb{P}_x$ and $\epsilon_{t+1:n}$ sequence of vectors of independent Rademacher random variables in $\{\pm 1\}^d$.
  2. Define q_t as the maximizer of

$$\min_{v \in \mathcal{D}} \left\{ q^T A v - \max_{f \in \mathcal{F}} \left\{ \sum_{s \neq t}^{n} f(x_s)^T Y_s^{(t)} + f(x_t)^T v \right\} \right\},$$

     over $q \in \Delta_{\mathcal{A}}$, with $\mathcal{D} := \{-1, 1\}^d$.
  3. Predict $a_t \sim q_t$ and observe v_t.

- **End for**.

4.3.2 Statistical versus Approximation Error with Full-Feedback

In this section, we discuss an important characteristic of the framework we adopt. Specifically, by expanding the set of functions $\mathcal{F}$, the approximation error of the set of mappings from contexts to actions decreases (simply because we have a richer class). The statistical error, however, increases as our algorithm needs to compete with a larger set of "experts". Theorem 4.1 captures this trade-off by establishing how the regret bound depends on the set of functions $\mathcal{F}$. In this section, we further demonstrate this trade-off by explicitly bounding the regret of Algorithm 8 for two classes of functions: the finite class of functions $\mathcal{F}$ and the class $\mathcal{F}$ that contains all linear mappings from contexts to valuations.

As shown in Theorem 4.1, the upper-bound on the expected regret is given by

$$\mathbb{E}_{x_{1:n}} 2\mathscr{R}(\mathcal{F}, x_{1:n}),$$

where $\mathscr{R}(\mathcal{F}, x_{1:n})$ is defined as

$$\mathscr{R}(\mathcal{F}, x_{1:n}) = \mathbb{E}_{\epsilon_{1:n}} \sup_{f \in \mathcal{F}} \left[\sum_{s=1}^{n} f(x_s)^T \epsilon_s \right].$$

Note that this definition resembles the definition of Rademacher averages but it is defined for vector-valued functions. In what follows, we show how to upper-bound the expected regret when the class of functions $\mathcal{F}$ is finite and when the class $\mathcal{F}$ is characterized by all linear mappings from contexts to values.

Finite Case

Consider the case in which the set of functions $\mathcal{F} = \{f \mid f : \mathcal{X} \to \mathcal{A}\}$ is any arbitrary finite set of functions from contexts to actions. Then, the next result provides a bound on the expected regret in the full-feedback setting when the class of policies is finite.

Proposition 4.1. *Suppose* $|\mathcal{F}| < \infty$ *and denote by* $Var(f) := \mathbb{E}_x \left(\sum_{i=1}^{d} [f(x)]_i^2 \right)$*, for* $f \in \mathcal{F}$ *and let* $f^* := argmax Var(f)$*. Then,*

$$\mathbb{E}_{x_{1:n}}(\textbf{Regret}) \leqslant 4\sqrt{n \log(|\mathcal{F}|)\, Var(f^*)}, \tag{4.7}$$

whenever $n > d \log(|\mathcal{F}|)/(1.79^2)$*.*

Note that Proposition 4.1 provides a bound that depends on the distribution of the contexts. Specifically, the bound depends on (the maximum) variance of the random variable $f(x_s)^T \epsilon_s$, for $s = 1, \ldots, n$. Note that if we directly apply Massart finite lemma to $\mathcal{F}$, we obtain $d\sqrt{2n \log |\mathcal{F}|}$, which does not depend on the distribution of the side information. Moreover, the bound provided by Massart finite lemma is larger than the one provided in Proposition 4.1 whenever $d \geqslant 2$ and n is large enough (because $\mathrm{Var}(f^*)$ can be at most d). Finally, the dependence on n of our bound is tight according to Dani et al. (2007) and Bubeck et al. (2012).

Linear Case

Consider now the case in which customers' valuations are linear in the contexts, i.e. $v(x_t) = Mx_t \in [-1,1]^d$ for some matrix $M \in \mathbb{R}^{d\times N}$. In this case, each context characteristic contributes linearly to the valuation that the customer associates with each feature of the product. Thus, in this case, a natural way to model the class of functions $\mathcal{F}$ is the following:

$$\mathcal{F} = \{f : \mathcal{X} \to \mathcal{A} : f(x) = \tau(Mx)\}, \tag{4.8}$$

where $\tau : \mathbb{R}^d \to \{0,1\}^d$ is a function that maps customers' valuations to the decision maker's action space $\{0,1\}^d$ according to the sign of the i-th coordinate of Mx_t. Specifically, for each $i = 1,\dots,d$, the i-th coordinate of $\tau(Mx_t)$ is

$$[\tau(Mx)]_i = \begin{cases} 1, & \text{if } [Mx]_i \geqslant 0 \\ 0, & \text{if } [Mx]_i < 0. \end{cases}$$

Note that modeling $\mathcal{F}$ in this way allows us to focus only on the customers' preferences. In other words, the decision maker is going to recommend the i-th feature of the product only if the customer values it, i.e., only if the i-th entry of Mx_t is nonnegative. Importantly, the function τ depends on the choice of the matrix M.

Given this formulation for $\mathcal{F}$, the bound provided in Theorem 4.1 can be upper bounded as follows:

Proposition 4.2. *Denote by $[x_s]_i$ the i-th element of x_s for $s = 1,\dots,n$ and let $M \in \mathbb{R}^{d\times N}$ be a matrix where each column vector m_i of M is such that $\|m_i\|_2 \leqslant C$ for $i = 1,2,\dots,N$. Then*

$$\mathscr{R}(\mathcal{F}, x_{1:n}) \leqslant \sqrt{dn}\left(CN\sqrt{\max_{s,i}([x_s]_i)^2} + d\sqrt{2\log(2)}\right). \tag{4.9}$$

Note that Proposition 4.2 provides a bound conditional on the sequence of side information $x_{1:n}$. Since (4.9) does not take into account the randomness in the contexts, we first prove Proposition 4.2 and then provide a corollary that bounds directly the expected regret.

In order to prove Proposition 4.2, we use two lemmas. We first start with the following.

Lemma 4.2. *Suppose* $\{\phi_s\}, \{\psi_s\}, s = 1, \dots, n$ *are two sets of functions from some set* $\mathcal{M}$ *to* $\mathbb{R}^d$ *such that for each* s *and* $M, M' \in \mathcal{M}$, $\|\phi_s(M) - \phi_s(M')\|_1 \leqslant \|\psi_s(M) - \psi_s(M')\|_1$. *Then*

$$\mathbb{E}_{\epsilon_{1:n}}\left[\sup_M \sum_{s=1}^n \phi_s(M)^T \epsilon_s\right] \leqslant \mathbb{E}_{\epsilon_{1:n}}\left[\sup_M \sum_{s=1}^n \psi_s(M)^T \epsilon_s\right]. \tag{4.10}$$

We now state a useful corollary of Lemma 4.2 that we will use to bound (4.9).

Corollary 4.1. *Let* $\mathcal{V} = \{v \mid v : \mathcal{X} \to \mathbb{R}^d\}$ *and suppose* $\gamma : \mathbb{R}^d \to \mathbb{R}^d$ *satisfies* $\|\gamma(y) - \gamma(y')\|_1 \leqslant C\|y - y'\|_1$ *for all* $y, y' \in \mathbb{R}^d$ *and some* $C > 0$. *Then*

$$\mathbb{E}_{\epsilon_{1:n}}\left[\sup_v \sum_{s=1}^n \gamma(v(x_s))^T \epsilon_s\right] \leqslant C\mathbb{E}_{\epsilon_{1:n}}\left[\sup_v \sum_{s=1}^n v(x_s)^T \epsilon_s\right]. \tag{4.11}$$

Now remember that, given the setting in the linear case (4.8), the quantity that we want to bound is

$$\mathbb{E}_{\epsilon_{1:n}}\left[\sup_M \sum_{s=1}^n \tau(Mx_s)^T \epsilon_s\right].$$

In order to do this, we define the following function

$$g(x) \coloneqq Mx + \delta \begin{pmatrix} \mathbb{1}\{[Mx]_1 \geqslant 0\} \\ \mathbb{1}\{[Mx]_2 \geqslant 0\} \\ \vdots \\ \mathbb{1}\{[Mx]_d \geqslant 0\} \end{pmatrix}$$

for some constant $\delta > 0$ to be chosen later. The reason we define g is that $\tau(Mx)$ does not satsify the condition of Corollary 4.1 (because of the discontinuity at zero). On the contrary, the function g shifts the nonnegative entries of Mx so that they are at least δ away from zero, allowing us to prove this condition for $\tau(g(\cdot))$. Clearly, given x_s, $\tau(Mx_s)^T\epsilon_s \stackrel{d}{=} \tau(g(x_s))^T\epsilon_s$ for any s, where the equality holds in distribution. Thus, conditional on $x_{1:n}$, it is enough to bound

$$\mathbb{E}_{\epsilon_{1:n}}\left[\sup_M \sum_{s=1}^n \tau(g(x_s))^T\epsilon_s\right].$$

Now, note that for each s and g, g', we have

$$\left|[\tau(g(x_s))]_j - [\tau(g'(x_s))]_j\right| \leqslant \frac{1}{\delta}\left|[g(x_s)]_j - [g'(x_s))]_j\right| \tag{4.12}$$

for each coordinate $j = 1, \ldots, d$. The reason equation (4.12) holds is the following: if $[g(x_s)]_j$ and $[g'(x_s)]_j$ have the same sign, the left-hand side is zero while the right-hand side is nonnegative; if $[g(x_s)]_j$ and $[g'(x_s)]_j$ have opposite signs, the left-hand side is one while the right-hand side is at least one (because $\delta \leqslant |[g(x_s)]_j - [g'(x_s))]_j|$, given the definition of g). Therefore, $\tau(g(\cdot))$ satisfies the conditions of Corollary 4.1 with $C = 1/\delta$, so that

$$\begin{aligned}\mathbb{E}_{\epsilon_{1:n}}\left[\sup_M \sum_{s=1}^n \tau(g(x_s))^T\epsilon_s\right] &\leqslant \frac{1}{\delta}\left(\mathbb{E}_{\epsilon_{1:n}}\left[\sup_M \sum_{s=1}^n g(x_s)^T\epsilon_s\right]\right) \\ &\stackrel{(a)}{\leqslant} \frac{1}{\delta}\left(\mathbb{E}_{\epsilon_{1:n}}\left[\sup_M \sum_{s=1}^n (Mx_s)^T\epsilon_s\right] + \mathbb{E}_{\epsilon_{1:n}}\left[\sup_{y_s \in \{0,\delta\}^d} \sum_{s=1}^n y_s^T\epsilon_s\right]\right),\end{aligned}$$

where (a) follows from splitting the supremum into two parts and taking the supremum in the second term over all possible vectors. We can now bound the second term on the right-hand side by Massart finite lemma, yielding the upper bound $d\delta\sqrt{2n\log(2^d)}$, while for the first term we can use the following lemma.

Lemma 4.3. *Suppose $v(x) = Mx$, with $M \in \mathbb{R}^{d\times N}$ and each column vector m_i of M is such that $\|m_i\|_2 \leqslant C$ for $i = 1, 2, \ldots, N$. Then*

$$\mathbb{E}_{\epsilon_{1:n}}\left[\sup_M \sum_{s=1}^{n} (Mx_s)^T \epsilon_s\right] \leqslant CN\sqrt{dn \max_{s,i}([x_s]_i)^2}. \tag{4.13}$$

Lemma 4.3 and Massart finite lemma then show that for any $\delta > 0$, we have

$$\mathbb{E}_{\epsilon_{1:n}}\left[\sup_M \sum_{s=1}^{n} \tau(g(x_s))^T \epsilon_s\right] \leqslant \frac{1}{\delta} CN\sqrt{dn \max_{s,i}([x_s]_i)^2} + d\sqrt{2\log(2)nd},$$

where C is as in Lemma 4.3. Now, choosing $\delta = 1$ (so that we can keep the dependence on the contexts), we obtain the desired bound in (4.9). Remember that the bound in Proposition 4.2 does not take into account the randomness in the contexts. The next corollary provides a bound on the expected regret in the case in which the entries are drawn i.i.d. from some distribution.

Corollary 4.2. *Suppose that the hypotheses of Proposition 4.2 hold true and that each entry of x_s is drawn i.i.d. from some distribution F_x on $[0, 1]$ with mean η. Then*

$$\mathbb{E}_{x_{1:n}}(\textbf{Regret}) \leqslant 2\sqrt{dn}\left(CN\left(\sqrt{\frac{\log(nN)}{2}} + \eta\right) + d\sqrt{2\log(2)}\right). \tag{4.14}$$

4.3.3 Computation of Algorithm 8

In this section, we provide some details about the optimization step in Algorithm 8. Note that, if the value of the empirical risk minimization (ERM) can be computed, the optimization problem to be solved, at each time period t, is the following:

$$\sup_{q\in\Delta_{\mathcal{A}}} \min_{v\in\mathcal{D}} \left\{ q^T Av - \max_{f\in\mathcal{F}} \left\{ \sum_{s\neq t}^{n} f(x_s)^T Y_s^{(t)} + f(x_t)^T v \right\} \right\}. \tag{4.15}$$

Note that this maximization step in Algorithm 8 can be computed in time $O(2^d)$, when we have 2^d accesses to a value optimization oracle. The proof is provided in Section 4.6.

Lemma 4.4. *Computing the maximizer of* (4.15) *can be done in time* $O(2^d)$ *and with* 2^d *accesses to a value optimization oracle.*

4.4 Bandit Feedback

In this section, we study a setting in which the decision maker, at each time t, observes the valuation of the recommended product a_t for customer x_t, i.e., at the end of round t, the decision maker observes $a_t^T v_t$. Specifically, let $\tilde{v}_t = Av_t \in [-d, d]^{2^d}$ denote the vector of all possible valuations at time t. Then, at each round $t \in \{1, \ldots, n\}$, the platform observes side information $x_t \in \mathcal{X}$ drawn from $\mathbb{P}_x$ and chooses a distribution $q_t \in \Delta_{\mathcal{A}}$. The adversary then chooses $v_t \in [-1, 1]^d$ and, finally, the decision maker draws an action $\hat{a}_t \sim q_t$ and observes the bandit feedback

$$\tilde{v}_t(\hat{a}_t) = e_{\hat{a}_t}^T \tilde{v}_t. \tag{4.16}$$

Note that the platform does not observe the valuation given by recommending other possible products (or actions). Also, remember that as we mentioned in Section 4.2, an important notation that we use is $e_{f(x)}$. This (abuse of) notation represents the vector whose components are all zero except for the index corresponding to the vector $f(x) \in \mathcal{A}$.

In this bandit feedback setting, we also augment the framework by adding some random state X_t drawn from some known distribution that can depend on q_t, $\hat{a}_t$ and $\tilde{v}_t(\hat{a}_t)$. Specifically, the bandit information obtained at round t is a tuple

$$I_t := I_t(x_t, q_t, \hat{a}_t, v_t, X_t) = (x_t, q_t, \hat{a}_t, \tilde{v}_t(\hat{a}_t), X_t),$$

where x_t is observed before q_t is chosen and $\tilde{v}_t(\hat{a}_t)$ is revealed. The random variable X_t will be constructed at the end of each round t and will be used to define an estimator for $\tilde{v}_t$. Now, similarly to the full-feedback setting, we define a bandit-information relaxation $\mathbf{Rel}^{\mathrm{BF}}(\cdot)$ to be a function that maps $(I_1, \ldots, I_t)$ to a real number, for any

$t \in \{1, \dots, n\}$. We say that a bandit-information relaxation $\mathbf{Rel}^{\mathrm{BF}}(\cdot)$ is **admissible** if for any $t \in \{1, \dots, n\}$ and any $(I_1, \dots, I_t)$,

$$\mathbb{E}_{x_t} \sup_{q_t} \min_{\tilde{v}_t} \mathbb{E}_{\hat{a}_t \sim q_t, X_t} \left\{ \tilde{v}_t(\hat{a}_t) + \mathbf{Rel}^{\mathrm{BF}}(I_{1:t-1}, I_t(x_t, q_t, \hat{a}_t, v_t, X_t)) \right\} \geqslant \mathbf{Rel}^{\mathrm{BF}}(I_{1:t-1}), \tag{4.17}$$

and for all $x_{1:n}$, $\tilde{v}_{1:n}$ and $q_{1:n}$

$$\mathbb{E}_{\hat{a}_{1:n} \sim q_{1:n}, X_{1:n}} \mathbf{Rel}^{\mathrm{BF}}(I_{1:n}) \leqslant - \sup_{f \in \mathcal{F}} \sum_{t=1}^{n} e_{f(x_t)}^T \tilde{v}_t. \tag{4.18}$$

Any randomized strategy $(q_t)_{t=1}^n$ that satisfies (4.17) and (4.18) is called an **admissible strategy**. Similarly to what we did in the full-feedback setting, we can prove the following lemma.

Lemma 4.5. *Let* $\mathbf{Rel}^{\mathrm{BF}}(\cdot)$ *be an admissible full-information relaxation and* $(q_t)_{t=1}^n$ *be an admissible strategy. Then, for any* $v_{1:n}$

$$\mathbb{E}_{x_{1:n}}(\mathbf{Regret}) \leqslant -\mathbf{Rel}^{\mathrm{BF}}(\varnothing).$$

Now, since at each time t, $\tilde{v}_t$ is not fully revealed, let us define an unbiased estimator for each possible reward vector $\tilde{v}_t$. We do this as follows. Consider a random variable X_t which we construct at the end of each round by conditioning on $\hat{a}_t$:

$$X_t = \begin{cases} 1 & \text{with probability } \frac{\tilde{v}_t(\hat{a}_t)}{L q_t(\hat{a}_t)} \\ 0 & \text{with the remaining probability} \end{cases} \tag{4.19}$$

where L is a constant to be specified later such that $L \geqslant d2^d$. Define the set $\mathcal{L} := \{Le_i : i \in [2^d]\} \cup \{\mathbf{0}\}$. Then, based on the information I_t, we construct an unbiased estimate for $\tilde{v}_t$ as follows:

$$\hat{v}_t = L X_t e_{\hat{a}_t} \in \mathcal{L}. \tag{4.20}$$

Note that for any $a \in \{0,1\}^d$:

$$\mathbb{E}\left[\hat{v}_t(a)\right] = \mathbb{E}\left[e_a^T \hat{v}_t\right] = \mathbb{E}\left[e_a^T L X_t e_{\hat{a}_t}\right] = \mathbb{E}\left[e_a^T e_{\hat{a}_t} L X_t\right] = \mathbb{P}(\hat{a}_t = a) L \frac{\tilde{v}_t(a)}{L q_t(a)} = \tilde{v}_t(a),$$

i.e., $\hat{v}_t$ is an unbiased estimate of $\tilde{v}_t$. Now, for any $t \in \{1, \ldots, n\}$, define $Z_t \in \{0, L\}$ to be a random variable which is L with probability $d2^d/L$ and 0 otherwise. Moreover, let $\epsilon_t \in \{-1, 1\}^{2^d}$ be a 2^d-dimensional vector of independent Rademacher random variables. Then, we can prove the following result.

Theorem 4.2. *The bandit-information relaxation*

$$\mathbf{Rel}^{\mathrm{BF}}(I_{1:t}) = \mathbb{E}_{(x,\epsilon,Z)_{(t+1):n}}\left[R(x_{1:t}, \hat{v}_{1:t}, (x, \epsilon, Z)_{(t+1):n})\right] \tag{4.21}$$

with

$$R(x_{1:t}, \hat{v}_{1:t}, (x, \epsilon, Z)_{(t+1):n}) = \inf_{f \in \mathcal{F}} \left\{ -\sum_{s=1}^{t} e_{f(x_s)}^T \hat{v}_s - \sum_{s=t+1}^{n} 2 e_{f(x_s)}^T \epsilon_s Z_s \right\} - \frac{(n-t) d 2^{d+1}}{L} \tag{4.22}$$

is admissible. An admissible randomized strategy for this relaxation is given by (4.31) *in Algorithm 9. The expected regret of the algorithm is upper bounded by*

$$2\mathbb{E}_{x_{1:n}} \mathbb{E}_{\epsilon_{1:n}} \mathbb{E}_{Z_{1:n}} \sup_{f \in \mathcal{F}} \left\{ \sum_{s=1}^{n} e_{f(x_s)}^T \epsilon_s Z_s \right\} + \frac{n d 2^{d+1}}{L}. \tag{4.23}$$

Theorem 4.2 establishes a bound on the regret of the relaxation-based Algorithm 9 in terms of Rademacher random variables of dimension 2^d. The dependence on $f(\cdot)$, however, is through $e_{f(\cdot)}$ which does not provide much intuition on how the structure of $f(\cdot)$ plays a role in this bound on the regret. We next present a corollary of this theorem that provides a bound in terms of the function $f(\cdot)$ itself. We make use of the following intermediary results to obtain this new bound.

Lemma 4.6. *Let $\{\phi_s\}_{s=1}^n$ be a set of functions from some set $\mathcal{M}$ to $(\mathbb{R}^p, \|\cdot\|_1)$ and $\{\psi_s\}_{s=1}^n$ be another set of functions from $\mathcal{M}$ to $(\mathbb{R}^d, \|\cdot\|_1)$. Suppose that for each s and $M, M' \in \mathcal{M}$, $\|\phi_s(M) - \phi_s(M')\|_1 \leqslant \|\psi_s(M) - \psi_s(M')\|_1$. Then*

$$\mathbb{E}_{\epsilon_{1:n}}\left[\sup_M \sum_{s=1}^n \phi_s(M)^T \epsilon_s\right] \leqslant \mathbb{E}_{\delta_{1:n}}\left[\sup_M \sum_{s=1}^n \psi_s(M)^T \delta_s\right], \tag{4.24}$$

where $\epsilon_{1:n}$ are p-dimensional Rademacher random variables and $\delta_{1:n}$ are d-dimensional Rademacher random variables.

We now state a useful corollary of Lemma 4.6.

Corollary 4.3. *Let $\mathcal{F} = \{f \mid f : \mathcal{X} \to \mathbb{R}^d\}$ and suppose $\gamma : (\mathbb{R}^d, \|\cdot\|_1) \to (\mathbb{R}^p, \|\cdot\|_1)$ satisfies $\|\gamma(y) - \gamma(y')\|_1 \leqslant C\|y - y'\|_1$ for all $y, y' \in \mathbb{R}^d$ and some $C > 0$. Then*

$$\mathbb{E}_{\epsilon_{1:n}}\left[\sup_f \sum_{s=1}^n \gamma(f(x_s))^T \epsilon_s\right] \leqslant C\mathbb{E}_{\delta_{1:n}}\left[\sup_f \sum_{s=1}^n f(x_s)^T \delta_s\right], \tag{4.25}$$

where $\epsilon_{1:n}$ are p-dimensional Rademacher random variables and $\delta_{1:n}$ are d-dimensional Rademacher random variables.

Given Corollary 4.3, we can now bound the regret of the relaxation-based Algorithm 9 as follows.

Let $\gamma(f(x_s)) = Z_s e_{f(x_s)}$ for given $Z_s \in \{0, L\}$ and x_s. Then, since $f(x_s) \in \{0,1\}^d$, we have that

$$Z_s\|e_{f(x_s)} - e_{f'(x_s)}\|_1 \leqslant 2Z_s\|f(x_s) - f'(x_s)\|_1,$$

because $\|e_{f(x_s)} - e_{f'(x_s)}\|_1$ is either 0 (if $f(x_s) = f'(x_s)$) or 2 (if $f(x_s) \neq f'(x_s)$), while $\|f(x_s) - f'(x_s)\|_1$ counts the number of different coordinates between $f(x_s)$ and $f'(x_s)$. Therefore, if $f(x_s) = f'(x_s)$, we have $0 \leqslant 0$, while if $f(x_s) \neq f'(x_s)$, then they differ in at least one coordinate, i.e. $1 \leqslant \|f(x_s) - f'(x_s)\|_1$. Thus, the conditions of Corollary

4.3 are satisfied (with $C = 2$) and we can conclude that

$$\mathbb{E}_{\epsilon_{1:n}}\left[\sup_f \sum_{s=1}^n e_{f(x_s)}^T \epsilon_s Z_s \mid Z_{1:n}, x_{1:n}\right] \leqslant 2\mathbb{E}_{\delta_{1:n}}\left[\sup_f \sum_{s=1}^n f(x_s)^T \delta_s Z_s \mid Z_{1:n}, x_{1:n}\right]$$

$$\iff \mathbb{E}_{x_{1:n}}\mathbb{E}_{Z_{1:n}}\mathbb{E}_{\epsilon_{1:n}}\left[\sup_f \sum_{s=1}^n e_{f(x_s)}^T \epsilon_s Z_s\right] \leqslant 2\mathbb{E}_{x_{1:n}}\mathbb{E}_{Z_{1:n}}\mathbb{E}_{\delta_{1:n}}\left[\sup_f \sum_{s=1}^n f(x_s)^T \delta_s Z_s\right]. \tag{4.26}$$

Given equation (4.26), we can state a useful corollary of Theorem 4.2 which we will use to bound the expected regret in the bandit feedback setting.

Corollary 4.4. *The bandit-information relaxation*

$$\mathbf{Rel}^{\mathrm{BF}}(I_{1:t}) = \mathbb{E}_{(x,\epsilon,Z)_{(t+1):n}}\left[R(x_{1:t}, \hat{v}_{1:t}, (x,\epsilon,Z)_{(t+1):n})\right] \tag{4.27}$$

with

$$R(x_{1:t}, \hat{v}_{1:t}, (x,\epsilon,Z)_{(t+1):n}) = \inf_{f\in\mathcal{F}}\left\{-\sum_{s=1}^t e_{f(x_s)}^T \hat{v}_s - \sum_{s=t+1}^n 2e_{f(x_s)}^T \epsilon_s Z_s\right\} - \frac{(n-t)d2^{d+1}}{L} \tag{4.28}$$

is admissible. An admissible randomized strategy for this relaxation is given by (4.31) *in Algorithm 9. The expected regret of the algorithm is upper bounded by*

$$4\mathbb{E}_{x_{1:n}}\mathbb{E}_{\delta_{1:n}}\mathbb{E}_{Z_{1:n}} \sup_{f\in\mathcal{F}}\left\{\sum_{s=1}^n f(x_s)^T \delta_s Z_s\right\} + \frac{nd2^{d+1}}{L}, \tag{4.29}$$

where $\delta_{1:n}$ are d-dimensional Rademacher random variables.

4.4.1 Online Algorithm for Bandit Feedback

In this section, we describe an algorithm whose regret is the bound provided in Theorem 4.2 and then discuss the computations involved in this algorithm in Subsection 4.4.3.

Algorithm 9

- **Input:** initialize a parameter L such that $L \geqslant d2^d$.
- **For** t from 1 to n **do**
  1. Observe x_t and draw $\rho_t = (x, \epsilon, Z)_{(t+1):n}$, where each x_s is drawn from $\mathbb{P}_x$, ϵ_s is a 2^d-dimensional Rademacher random variable and $Z_s \in \{0, L\}$ is L with probability $d2^d/L$ and 0 otherwise.
  2. Define $q_t^*(\rho_t)$ to be the maximizer of

$$\inf_{p_t \in \Delta'_{\mathcal{L}}} \mathbb{E}_{\hat{v} \sim p_t} \left[q^T \hat{v} + R(x_{1:t}, \hat{v}_{1:t-1}, \hat{v}, \rho_t) \right] \tag{4.30}$$

     over $q \in \Delta_{\mathcal{A}}$ with $\Delta'_{\mathcal{L}} := \{p \in \Delta_{\mathcal{L}} : p(i) \leqslant d/L \ \forall i \in [2^d]\}$ and set

$$q_t(\rho_t) = \left(1 - \frac{2^d}{L}\right) q_t^*(\rho_t) + \frac{1}{L}\mathbf{1}. \tag{4.31}$$

  3. Predict $\hat{a}_t \sim q_t(\rho_t)$ and observe $\tilde{v}_t(\hat{a}_t)$.
  4. Create an estimate $\hat{v}_t = LX_t e_{\hat{a}_t}$ as in (4.20), with X_t defined in (4.19) with q_t in that equation instantiated with $q_t(\rho_t)$.
- **End for**.

4.4.2 Statistical versus Approximation Error with Bandit Feedback

Similarly to Section 4.3.2, Theorem 4.2 captures the approximation-statistical error trade-off by establishing how the regret bound depends on the set of functions $\mathcal{F}$. As shown in Corollary 4.4, an upper-bound on the expected regret of Algorithm 9 is

given by

$$4\mathbb{E}_{x_{1:n}}\mathbb{E}_{\delta_{1:n}}\mathbb{E}_{Z_{1:n}}\sup_{f\in\mathcal{F}}\left\{\sum_{s=1}^{n} f(x_s)^T\delta_s Z_s\right\} + \frac{nd2^{d+1}}{L}. \tag{4.32}$$

In this section, we explicitly bound this quantity for two classes of functions: the finite class of functions $\mathcal{F}$ and the class $\mathcal{F}$ that contains all linear mappings from contexts to valuations. Then, the next result provides a bound on the expected regret in the bandit feedback setting when the class of policies is finite.

Finite Case

Similarly to the full-feedback setting, we first consider the case in which the set of functions $\mathcal{F} = \{f \mid f : \mathcal{X} \to \mathcal{A}\}$ is any arbitrary finite set of functions from contexts to actions.

Proposition 4.3. *Suppose* $|\mathcal{F}| < \infty$ *and denote by* $Var(f) \coloneqq \mathbb{E}_x\left(\sum_{i=1}^{d}[f(x)]_i^2\right)$, *for* $f \in \mathcal{F}$ *and let* $f^* \coloneqq argmax Var(f)$. *Then,*

$$\mathbb{E}_{x_{1:n}}(\textbf{Regret}) \leqslant 2^{14/3}\left(nd2^d\right)^{2/3}\left(\log(|\mathcal{F}|)\,Var(f^*)\right)^{1/3},$$

whenever $n \geqslant 4(d2^d)^2\log(|\mathcal{F}|)\,Var(f^*)$.

Note that, again, similarly to the full-feedback setting, Proposition 4.3 provides a bound that depends on the distribution of the contexts. The optimal dependence on n of the regret in this finite case (when $|\mathcal{F}| < \infty$) is not yet settled. Specifically, the best current upper bound has $\widetilde{O}(n^{2/3})$ dependence on n (proved in Syrgkanis et al. (2016)), against the $\Omega(\sqrt{n})$ lower bound in Slivkins (2019).

Linear Case

Let us consider now the linear case framework, i.e.,

$$\mathcal{F} = \{f : \mathcal{X} \to \mathcal{A} : f(x) = \tau(Mx)\},$$

where $\tau : \mathbb{R}^d \to \{0,1\}^d$ is a function that maps customers' valuations to the decision maker's action space $\{0,1\}^d$ according to the sign of the i-th coordinate of Mx_t. Again, for each $i = 1, \ldots, d$, we have that

$$[\tau(Mx)]_i = \begin{cases} 1, & \text{if } [Mx]_i \geqslant 0 \\ 0, & \text{if } [Mx]_i < 0. \end{cases}$$

In Section 4.6, we show that the analogue of Lemma 4.3 holds in this bandit feedback setting, enabling us to further upper bound the bound provided by Corollary 4.4. The formal statement is provided next.

Proposition 4.4. *Denote by $[x_s]_i$ the i-th element of x_s for $s = 1, \ldots, n$ and let $M \in \mathbb{R}^{d \times N}$ be a matrix where each column vector m_i of M is such that $\|m_i\|_2 \leqslant C$ for $i = 1, 2, \ldots, N$. Then*

$$\mathbb{E}_{x_{1:n}}(\textbf{Regret}) \leqslant 2^{8/3} d (n 2^d)^{2/3} \left(CN \mathbb{E}_{x_{1:n}} \left[\sqrt{\max_{s,i} ([x_s]_i)^2} \right] + d\sqrt{2\log(2)} \right)^{2/3}, \quad (4.33)$$

whenever $n \geqslant d^3 (2^d)^2 \left(CN \mathbb{E}_{x_{1:n}} \left[\sqrt{\max_{s,i} ([x_s]_i)^2} \right] + d\sqrt{2\log(2)} \right)^2$.

Note that the bound in Proposition 4.4 depends on the expectation of the largest entry (in absolute value) of the sequence $x_{1:n}$. We extend this result in the next corollary by providing a bound on the expected regret in the case in which the entries are drawn i.i.d. from some distribution.

Corollary 4.5. *Suppose that the hypotheses of Proposition 4.4 hold true and that each entry of x_s is drawn i.i.d. from some distribution F_x on $[0,1]$ with mean η. Then*

$$\mathbb{E}_{x_{1:n}}(\textbf{Regret}) \leqslant 2^{8/3} d (n 2^d)^{2/3} \left(CN \left(\sqrt{\frac{\log(nN)}{2}} + \eta \right) + d\sqrt{2\log(2)} \right)^{2/3}, \quad (4.34)$$

whenever $n \geqslant d^3 (2^d)^2 \left(CN \mathbb{E}_{x_{1:n}} \left[\sqrt{\max_{s,i} ([x_s]_i)^2} \right] + d\sqrt{2\log(2)} \right)^2$.

4.4.3 Computation of Algorithm 9

In this section, we provide some details about the optimization step in Algorithm 9. Note that, if the value of the ERM can be computed, the optimization problem to be solved in the bandit feedback setting, at each time period t, is the following:

$$\sup_{q \in \Delta_{\mathcal{A}}} \inf_{p_t \in \Delta'_{\mathcal{L}}} \mathbb{E}_{\hat{v}_t \sim p_t} \left[q^T \hat{v}_t + R(x_{1:t}, \hat{v}_{1:t}, \rho_t) \right]. \tag{4.35}$$

Note that this maximization step in equation (4.30) can be computed in time $O(2^d)$, when we have $2^d + 1$ accesses to a value optimization oracle. The proof is provided in Section 4.6.

Lemma 4.7. *Computing the maximizer of the objective* (4.30) *for any given* ρ_t*, can be done in time* $O(2^d)$ *and with* $2^d + 1$ *accesses to a value optimization oracle.*

4.5 Conclusion

In this chapter, we consider a contextual bandit formulation for our online recommendation problem. Specifically, at each time, a decision-maker is faced with the problem of choosing which features of a given product to recommend. We develop a relaxation-based algorithm that can compete against any set of policies that map contexts to actions with either full-feedback or bandit feedback. We then apply this framework to two specific classes of policies (finite and linear) and establish how the regret scales with the class parameters, i.e., the size of the finite class and the sparsity of the linear class.

Avenues for future research include characterizing the regret bound for richer classes of policies. For instance, one can consider the case in which users belong to a set of types and types reside in a (social) network. The values of types that are connected in the social network are related to each other and the decision-maker's goal is to use the social network structure in order to develop a better online recommendation system.

4.6 Proofs

This section contains all the omitted proofs from the text.

Proof of Lemma 4.1

Let $(q_t)_{t=1}^n$ be an admissible strategy and define $\overline{\mathbf{Regret}} = -\mathbf{Regret}$. Then:

$$\begin{aligned}
\mathbb{E}_{x_{1:n}}(\overline{\mathbf{Regret}}) &\geqslant \inf_{v_{1:n}} \mathbb{E}_{x_{1:n}}(\overline{\mathbf{Regret}}) \\
&\overset{(a)}{=} \inf_{v_1} \inf_{v_2} \cdots \inf_{v_{n-1}} \inf_{v_n} \mathbb{E}_{x_{1:n}}(\overline{\mathbf{Regret}}) \\
&\overset{(b)}{\geqslant} \inf_{v_1} \inf_{v_2} \cdots \inf_{v_{n-1}} \mathbb{E}_{x_{1:n}}(\inf_{v_n} \overline{\mathbf{Regret}}) \\
&\overset{(c)}{=} \inf_{v_1} \inf_{v_2} \cdots \inf_{v_{n-1}} \mathbb{E}_{x_{1:(n-1)}} \mathbb{E}_{x_n}(\inf_{v_n} \overline{\mathbf{Regret}}) \\
&\geqslant \inf_{v_1} \inf_{v_2} \cdots \mathbb{E}_{x_{1:(n-1)}} \inf_{v_{n-1}} \mathbb{E}_{x_n}(\inf_{v_n} \overline{\mathbf{Regret}}) \\
&\vdots \\
&\geqslant \mathbb{E}_{x_1} \inf_{v_1} \mathbb{E}_{x_2} \inf_{v_2} \cdots \mathbb{E}_{x_{n-1}} \inf_{v_{n-1}} \mathbb{E}_{x_n}(\inf_{v_n} \overline{\mathbf{Regret}}),
\end{aligned}$$

where (a) follows from the fact that, for bounded f, $\inf_{(x,y)} f(x,y) = \inf_x \inf_y f(x,y)$; (b) follows from $\inf_{y\in\mathcal{Y}} \mathbb{E}[f(X,y)] \geqslant \mathbb{E}[\inf_{y\in\mathcal{Y}} f(X,y)]$ for any measurable f; and (c) follows from the independence of the contexts.

Now, by admissibility of $\mathbf{Rel}^{\mathrm{FF}}(\cdot)$, we have that

$$\mathbf{Rel}^{\mathrm{FF}}(I_{1:n}) \leqslant -\sup_{f\in\mathcal{F}} \sum_{t=1}^{n} f(x_t)^T v_t.$$

Therefore, we can further lower bound $\mathbb{E}_{x_{1:n}}(\overline{\mathbf{Regret}})$ as follows:

$$
\begin{aligned}
&\mathbb{E}_{x_{1:n}}(\overline{\mathbf{Regret}}) \\
&\geqslant \mathbb{E}_{x_1} \inf_{v_1} \mathbb{E}_{x_2} \inf_{v_2} \cdots \mathbb{E}_{x_{n-1}} \inf_{v_{n-1}} \mathbb{E}_{x_n} \inf_{v_n} \left(\sum_{t=1}^{n} q_t^T A v_t - \sup_{f \in \mathcal{F}} \sum_{t=1}^{n} f(x_t)^T v_t \right) \\
&\overset{(a)}{\geqslant} \mathbb{E}_{x_1} \inf_{v_1} \mathbb{E}_{x_2} \inf_{v_2} \cdots \mathbb{E}_{x_{n-1}} \inf_{v_{n-1}} \mathbb{E}_{x_n} \inf_{v_n} \left(\sum_{t=1}^{n} q_t^T A v_t + \mathbf{Rel}^{\mathrm{FF}}(I_{1:n}) \right) \\
&= \mathbb{E}_{x_1} \inf_{v_1} \mathbb{E}_{x_2} \inf_{v_2} \cdots \mathbb{E}_{x_{n-1}} \inf_{v_{n-1}} \mathbb{E}_{x_n} \inf_{v_n} \left(\mathbb{E}_{a_{1:n} \sim q_{1:n}} \left[\sum_{t=1}^{n} a_t^T v_t + \mathbf{Rel}^{\mathrm{FF}}(I_{1:n}) \right] \right) \\
&= \mathbb{E}_{x_1} \inf_{v_1} \mathbb{E}_{x_2} \inf_{v_2} \cdots \mathbb{E}_{x_{n-1}} \inf_{v_{n-1}} \mathbb{E}_{x_n} \inf_{v_n} \left(\mathbb{E}_{a_{1:(n-1)}} \mathbb{E}_{a_n \sim q_n} \left[\sum_{t=1}^{n} a_t^T v_t + \mathbf{Rel}^{\mathrm{FF}}(I_{1:n}) \right] \right) \\
&\geqslant \mathbb{E}_{x_1} \inf_{v_1} \mathbb{E}_{x_2} \inf_{v_2} \cdots \mathbb{E}_{x_{n-1}} \inf_{v_{n-1}} \mathbb{E}_{x_n} \mathbb{E}_{a_{1:(n-1)}} \left(\inf_{v_n} \mathbb{E}_{a_n \sim q_n} \left[\sum_{t=1}^{n} a_t^T v_t + \mathbf{Rel}^{\mathrm{FF}}(I_{1:n}) \right] \right) \\
&\overset{(b)}{=} \mathbb{E}_{x_1} \inf_{v_1} \mathbb{E}_{x_2} \inf_{v_2} \cdots \mathbb{E}_{x_{n-1}} \inf_{v_{n-1}} \mathbb{E}_{a_{1:(n-1)}} \mathbb{E}_{x_n} \left(\inf_{v_n} \mathbb{E}_{a_n \sim q_n} \left[\sum_{t=1}^{n} a_t^T v_t + \mathbf{Rel}^{\mathrm{FF}}(I_{1:n}) \right] \right) \\
&\vdots \\
&\overset{(c)}{\geqslant} \mathbb{E}_{x_1} \inf_{v_1} \mathbb{E}_{a_1 \sim q_1} \mathbb{E}_{x_2} \inf_{v_2} \mathbb{E}_{a_2 \sim q_2} \cdots \mathbb{E}_{x_n} \inf_{v_n} \mathbb{E}_{a_n \sim q_n} \left[\sum_{t=1}^{n} a_t^T v_t + \mathbf{Rel}^{\mathrm{FF}}(I_{1:n}) \right] \\
&= \mathbb{E}_{x_1} \inf_{v_1} \mathbb{E}_{a_1 \sim q_1} \mathbb{E}_{x_2} \inf_{v_2} \mathbb{E}_{a_2 \sim q_2} \cdots \mathbb{E}_{x_n} \inf_{v_n} \mathbb{E}_{a_n \sim q_n} \left[\sum_{t=1}^{n-1} a_t^T v_t + a_n^T v_n + \mathbf{Rel}^{\mathrm{FF}}(I_{1:n}) \right] \\
&= \mathbb{E}_{x_1} \inf_{v_1} \mathbb{E}_{a_1 \sim q_1} \mathbb{E}_{x_2} \inf_{v_2} \mathbb{E}_{a_2 \sim q_2} \cdots \mathbb{E}_{x_{n-1}} \inf_{v_{n-1}} \mathbb{E}_{a_{n-1} \sim q_{n-1}} \Bigg[\sum_{t=1}^{n-1} a_t^T v_t + \\
&\quad + \mathbb{E}_{x_n} \inf_{v_n} \mathbb{E}_{a_n \sim q_n} \{ a_n^T v_n + \mathbf{Rel}^{\mathrm{FF}}(I_{1:n}) \} \Bigg] \\
&\overset{(d)}{\geqslant} \mathbb{E}_{x_1} \inf_{v_1} \mathbb{E}_{a_1 \sim q_1} \mathbb{E}_{x_2} \inf_{v_2} \mathbb{E}_{a_2 \sim q_2} \cdots \mathbb{E}_{x_{n-1}} \inf_{v_{n-1}} \mathbb{E}_{a_{n-1} \sim q_{n-1}} \left[\sum_{t=1}^{n-1} a_t^T v_t + \mathbf{Rel}^{\mathrm{FF}}(I_{1:n-1}) \right],
\end{aligned}
$$

where (a) holds by admissibility, (b) holds by Fubini Theorem and (d) holds by

equation (4.3), since $(q_t)_{t=1}^n$ is admissible. Finally, proceeding iteratively from step (c), we obtain the final bound

$$\mathbb{E}_{x_{1:n}}(\overline{\mathbf{Regret}}) \geqslant \mathbf{Rel}^{\mathrm{FF}}(\varnothing),$$

i.e. $\mathbb{E}_{x_{1:n}}(\mathbf{Regret}) \leqslant -\mathbf{Rel}^{\mathrm{FF}}(\varnothing)$. This completes the proof. ■

Proof of Theorem 4.1

Admissibility: initial condition. For any $v_{1:n}, x_{1:n}$ it holds that

$$-\sup_{f\in\mathcal{F}} \sum_{s=1}^{n} f(x_s)^T v_s = \inf_{f\in\mathcal{F}} -\sum_{s=1}^{n} f(x_s)^T Y_s^{(n)} = \mathbf{Rel}^{\mathrm{FF}}(I_{1:n}).$$

Admissibility: recursion. Let $\mathcal{D} \coloneqq \{-1, 1\}^d$ and let $\epsilon_s \in \{\pm 1\}^d$ denote a vector of independent Rademacher random variables. We will now reason conditionally on x_t. Let us denote by

$$\rho_t \coloneqq (x_{(t+1):n}, \epsilon_{(t+1):n}),$$

a draw of independent covariates from $\mathbb{P}_x$ and Rademacher random variables for the "future rounds". Define the randomized prediction algorithm as:

$$q_t(\rho_t) \coloneqq \operatorname*{argsup}_{q\in\Delta_{\mathcal{A}}} \inf_{v\in\mathcal{D}} \left\{ q^T A v + \inf_{f\in\mathcal{F}} \left\{ -\sum_{s\neq t}^{n} f(x_s)^T Y_s^{(t)} - f(x_t)^T v \right\} \right\},$$

where $\Delta_{\mathcal{A}}$ denotes the set of distributions over $\mathcal{A}$. Let $q_t \coloneqq \mathbb{E}_{\rho_t}[q_t(\rho)]$. Then:

$$\min_{v_t \in [-1,1]^d} \mathbb{E}_{a_t \sim q_t} \left\{ a_t^T v_t + \mathbf{Rel}^{\mathrm{FF}}(I_{1:t}) \right\}$$

$$\overset{(a)}{=} \min_{v_t \in [-1,1]^d} \left\{ q_t^T A v_t + \mathbf{Rel}^{\mathrm{FF}}(I_{1:(t-1)}, I_t(x_t, q_t, a_t, v_t)) \right\}$$

$$\overset{(b)}{=} \min_{v_t \in [-1,1]^d} \left\{ q_t^T A v_t + \mathbb{E}_{\rho_t} \inf_{f \in \mathcal{F}} \left\{ -\sum_{s \neq t}^{n} f(x_s)^T Y_s^{(t)} - f(x_t)^T v_t \right\} \right\}$$

$$\overset{(c)}{=} \inf_{v_t \in \mathcal{D}} \left\{ q_t^T A v_t + \mathbb{E}_{\rho_t} \inf_{f \in \mathcal{F}} \left\{ -\sum_{s \neq t}^{n} f(x_s)^T Y_s^{(t)} - f(x_t)^T v_t \right\} \right\}$$

$$\overset{(d)}{\geqslant} \mathbb{E}_{\rho_t} \inf_{v_t \in \mathcal{D}} \left\{ q_t(\rho_t)^T A v_t + \inf_{f \in \mathcal{F}} \left\{ -\sum_{s \neq t}^{n} f(x_s)^T Y_s^{(t)} - f(x_t)^T v_t \right\} \right\}$$

$$\overset{(e)}{=} \mathbb{E}_{\rho_t} \sup_{q \in \Delta_{\mathcal{A}}} \inf_{v_t \in \mathcal{D}} \left\{ q^T A v_t + \inf_{f \in \mathcal{F}} \left\{ -\sum_{s \neq t}^{n} f(x_s)^T Y_s^{(t)} - f(x_t)^T v_t \right\} \right\}$$

$$\overset{(f)}{=} \mathbb{E}_{\rho_t} \sup_{q \in \Delta_{\mathcal{A}}} \inf_{p_t} \mathbb{E}_{v_t \sim p_t} \left\{ q^T A v_t + \inf_{f \in \mathcal{F}} \left\{ -\sum_{s \neq t}^{n} f(x_s)^T Y_s^{(t)} - f(x_t)^T v_t \right\} \right\}$$

$$\overset{(g)}{=} \mathbb{E}_{\rho_t} \inf_{p_t} \sup_{q \in \Delta_{\mathcal{A}}} \mathbb{E}_{v_t \sim p_t} \left\{ q^T A v_t + \inf_{f \in \mathcal{F}} \left\{ -\sum_{s \neq t}^{n} f(x_s)^T Y_s^{(t)} - f(x_t)^T v_t \right\} \right\}$$

$$= \mathbb{E}_{\rho_t} \inf_{p_t} \sup_{q \in \Delta_{\mathcal{A}}} \left\{ q^T A \mathbb{E}_{v_t \sim p_t}[v_t] + \mathbb{E}_{v_t \sim p_t} \inf_{f \in \mathcal{F}} \left\{ -\sum_{s \neq t}^{n} f(x_s)^T Y_s^{(t)} - f(x_t)^T v_t \right\} \right\}$$

$$= \mathbb{E}_{\rho_t} \inf_{p_t} \left\{ \max_{j \in [2^d]} e_j^T A \mathbb{E}_{v_t \sim p_t}[v_t] + \mathbb{E}_{v_t \sim p_t} \inf_{f \in \mathcal{F}} \left\{ -\sum_{s \neq t}^{n} f(x_s)^T Y_s^{(t)} - f(x_t)^T v_t \right\} \right\}$$

$$\overset{(h)}{=} \mathbb{E}_{\rho_t} \inf_{p_t} \left\{ \mathbb{E}_{v_t \sim p_t} \inf_{f \in \mathcal{F}} \left\{ -\sum_{s \neq t}^{n} f(x_s)^T Y_s^{(t)} + \max_{j \in [2^d]} e_j^T A \mathbb{E}_{v_t \sim p_t}[v_t] - f(x_t)^T v_t \right\} \right\},$$

where (a) follows from the fact that $\mathbb{E}_{a_t \sim q_t} a_t^T v_t = q_t^T A v_t$; (b) follows from the definition of $\mathbf{Rel}^{\mathrm{FF}}(\cdot)$; (c) follows from the fact that, since the objective is concave in v_t, the

infimum will be achieved at a vertex of the set $[-1,1]^d$; (d) follows from the inequality $\inf_{y\in\mathcal{Y}} \mathbb{E}[f(X,y)] \geqslant \mathbb{E}[\inf_{y\in\mathcal{Y}} f(X,y)]$ for any measurable f; (e) follows from the definition of $q_t(\rho_t)$; (f) follows from the fact that the infimum will be achieved at a vertex of $[-1,1]^d$ and we are taking the infimum over all distributions p_t on $\{-1,1\}^d$; (g) follows from the Minimax Theorem (since the objective is linear in both q and p_t) and (h) holds because $\max_{j\in[2^d]} e_j^T A\mathbb{E}_{v_t\sim p_t}[v_t]$ is a constant.

The rest of the lower bounds will be derived conditionally on ρ_t. In particular, for δ_t one-dimensional Rademacher random variable, we have

$$\begin{aligned}
&\inf_{p_t}\left\{\mathbb{E}_{v_t\sim p_t}\inf_{f\in\mathcal{F}}\left\{-\sum_{s\neq t}^{n} f(x_s)^T Y_s^{(t)} + \max_{j\in[2^d]} e_j^T A\mathbb{E}_{v_t\sim p_t}[v_t] - f(x_t)^T v_t\right\}\right\} \\
&\overset{(a)}{\geqslant} \inf_{p_t}\left\{\mathbb{E}_{v_t\sim p_t}\inf_{f\in\mathcal{F}}\left\{-\sum_{s\neq t}^{n} f(x_s)^T Y_s^{(t)} + f(x_t)^T\mathbb{E}_{v_t\sim p_t}[v_t] - f(x_t)^T v_t\right\}\right\} \\
&\overset{(b)}{\geqslant} \inf_{p_t}\left\{\mathbb{E}_{v_t\sim p_t, v_t'\sim p_t}\inf_{f\in\mathcal{F}}\left\{-\sum_{s\neq t}^{n} f(x_s)^T Y_s^{(t)} + f(x_t)^T(v_t' - v_t)\right\}\right\} \\
&\overset{(c)}{=} \inf_{p_t}\left\{\mathbb{E}_{v_t\sim p_t, v_t'\sim p_t}\mathbb{E}_{\delta_t}\inf_{f\in\mathcal{F}}\left\{-\sum_{s\neq t}^{n} f(x_s)^T Y_s^{(t)} + \delta_t f(x_t)^T(v_t' - v_t)\right\}\right\},
\end{aligned}$$

where (a) follows from the fact that $\max_{j\in[2^d]} e_j^T A\mathbb{E}_{v_t\sim p_t}[v_t] \geqslant f(x_t)^T\mathbb{E}_{v_t\sim p_t}[v_t]$; (b) follows from $\inf_{y\in\mathcal{Y}} \mathbb{E}[f(X,y)] \geqslant \mathbb{E}[\inf_{y\in\mathcal{Y}} f(X,y)]$ and (c) holds because $\mathbb{E}[g(X - X')] = \mathbb{E}[g(X' - X)]$, for any X, X' i.i.d. Now notice that the expression inside the

first infimum (in the latter expression) can be decomposed as follows:

$$\mathbb{E}_{v_t\sim p_t, v_t'\sim p_t}\mathbb{E}_{\delta_t}\inf_{f\in\mathcal{F}}\left\{-\sum_{s\neq t}^{n} f(x_s)^T Y_s^{(t)} + \delta_t f(x_t)^T(v_t' - v_t)\right\}$$

$$= \mathbb{E}_{v_t\sim p_t, v_t'\sim p_t}\mathbb{E}_{\delta_t}\inf_{f\in\mathcal{F}}\left\{-\frac{1}{2}\sum_{s\neq t}^{n} f(x_s)^T Y_s^{(t)} + \delta_t f(x_t)^T v_t' - \frac{1}{2}\sum_{s\neq t}^{n} f(x_s)^T Y_s^{(t)} - \delta_t f(x_t)^T v_t\right\}$$

$$\geqslant \mathbb{E}_{v_t\sim p_t, v_t'\sim p_t}\mathbb{E}_{\delta_t}\left\{\inf_{f\in\mathcal{F}}\left\{-\frac{1}{2}\sum_{s\neq t}^{n} f(x_s)^T Y_s^{(t)} + \delta_t f(x_t)^T v_t'\right\} + \right.$$

$$\left. + \inf_{f\in\mathcal{F}}\left\{-\frac{1}{2}\sum_{s\neq t}^{n} f(x_s)^T Y_s^{(t)} - \delta_t f(x_t)^T v_t\right\}\right\}$$

$$= \mathbb{E}_{v_t\sim p_t, v_t'\sim p_t}\left\{\mathbb{E}_{\delta_t}\inf_{f\in\mathcal{F}}\left\{-\frac{1}{2}\sum_{s\neq t}^{n} f(x_s)^T Y_s^{(t)} + \delta_t f(x_t)^T v_t'\right\} + \right.$$

$$\left. + \mathbb{E}_{\delta_t}\inf_{f\in\mathcal{F}}\left\{-\frac{1}{2}\sum_{s\neq t}^{n} f(x_s)^T Y_s^{(t)} - \delta_t f(x_t)^T v_t\right\}\right\}$$

$$= \mathbb{E}_{v_t'\sim p_t}\mathbb{E}_{\delta_t}\inf_{f\in\mathcal{F}}\left\{-\frac{1}{2}\sum_{s\neq t}^{n} f(x_s)^T Y_s^{(t)} + \delta_t f(x_t)^T v_t'\right\} +$$

$$+ \mathbb{E}_{v_t\sim p_t}\mathbb{E}_{\delta_t}\inf_{f\in\mathcal{F}}\left\{-\frac{1}{2}\sum_{s\neq t}^{n} f(x_s)^T Y_s^{(t)} - \delta_t f(x_t)^T v_t\right\}$$

$$\overset{(a)}{=} \mathbb{E}_{v_t'\sim p_t}\mathbb{E}_{\delta_t}\inf_{f\in\mathcal{F}}\left\{-\frac{1}{2}\sum_{s\neq t}^{n} f(x_s)^T Y_s^{(t)} + \delta_t f(x_t)^T v_t'\right\} +$$

$$+ \mathbb{E}_{v_t\sim p_t}\mathbb{E}_{\delta_t}\inf_{f\in\mathcal{F}}\left\{-\frac{1}{2}\sum_{s\neq t}^{n} f(x_s)^T Y_s^{(t)} + \delta_t f(x_t)^T v_t\right\}$$

$$= 2\mathbb{E}_{v_t\sim p_t}\mathbb{E}_{\delta_t}\inf_{f\in\mathcal{F}}\left\{-\frac{1}{2}\sum_{s\neq t}^{n} f(x_s)^T Y_s^{(t)} + \delta_t f(x_t)^T v_t\right\}$$

$$= \mathbb{E}_{v_t\sim p_t}\mathbb{E}_{\delta_t}\inf_{f\in\mathcal{F}}\left\{-\sum_{s\neq t}^{n} f(x_s)^T Y_s^{(t)} + 2\delta_t f(x_t)^T v_t\right\},$$

where (a) holds because δ_t and $-\delta_t$ have the same distribution. Thus,

$$\inf_{p_t} \left\{ \mathbb{E}_{v_t \sim p_t, v_t' \sim p_t} \mathbb{E}_{\delta_t} \inf_{f \in \mathcal{F}} \left\{ -\sum_{s \neq t}^{n} f(x_s)^T Y_s^{(t)} + \delta_t f(x_t)^T (v_t' - v_t) \right\} \right\}$$

$$\geqslant \inf_{p_t} \left\{ \mathbb{E}_{v_t \sim p_t} \mathbb{E}_{\delta_t} \inf_{f \in \mathcal{F}} \left\{ -\sum_{s \neq t}^{n} f(x_s)^T Y_s^{(t)} + 2\delta_t f(x_t)^T v_t \right\} \right\},$$

and so the overall bound is now given by

$$\min_{v_t \in [-1,1]^d} \mathbb{E}_{a_t \sim q_t} \left\{ a_t^T v_t + \mathbf{Rel}^{\mathrm{FF}}(I_{1:t}) \right\}$$

$$\geqslant \mathbb{E}_{\rho_t} \inf_{p_t} \left\{ \mathbb{E}_{v_t \sim p_t} \inf_{f \in \mathcal{F}} \left\{ -\sum_{s \neq t}^{n} f(x_s)^T Y_s^{(t)} + \max_{j \in [2^d]} e_j^T A \mathbb{E}_{v_t \sim p_t}[v_t] - f(x_t)^T v_t \right\} \right\}$$

$$\geqslant \mathbb{E}_{\rho_t} \inf_{p_t} \left\{ \mathbb{E}_{v_t \sim p_t} \mathbb{E}_{\delta_t} \inf_{f \in \mathcal{F}} \left\{ -\sum_{s \neq t}^{n} f(x_s)^T Y_s^{(t)} + 2\delta_t f(x_t)^T v_t \right\} \right\}$$

$$\overset{(a)}{=} \mathbb{E}_{\rho_t} \inf_{p_t} \left\{ \mathbb{E}_{v_t \sim p_t} \mathbb{E}_{\epsilon_t} \inf_{f \in \mathcal{F}} \left\{ -\sum_{s \neq t}^{n} f(x_s)^T Y_s^{(t)} + 2 f(x_t)^T \epsilon_t \right\} \right\}$$

$$= \mathbb{E}_{\rho_t} \mathbb{E}_{\epsilon_t} \inf_{f \in \mathcal{F}} \left\{ -\sum_{s \neq t}^{n} f(x_s)^T Y_s^{(t)} + 2 f(x_t)^T \epsilon_t \right\},$$

where (a) follows from the fact that $\delta_t v_t$ is a d-dimensional Rademacher for $v_t \in \mathcal{D}$. Now note that the above lower bound holds for any x_t. Therefore, we can take the expectation on both sides with respect to x_t, so that

$$\mathbb{E}_{x_t} \min_{v_t \in [-1,1]^d} \mathbb{E}_{a_t \sim q_t} \left\{ a_t^T v_t + \mathbf{Rel}^{\mathrm{FF}}(I_{1:t}) \right\} \geqslant \mathbb{E}_{x_{t:n}, \epsilon_{t:n}} \inf_{f \in \mathcal{F}} \left\{ -\sum_{s \neq t}^{n} f(x_s)^T Y_s^{(t)} + 2 f(x_t)^T \epsilon_t \right\}$$

$$= \mathbf{Rel}^{\mathrm{FF}}(I_{1:(t-1)}).$$

This proves admissibility.

Regret bound. The final bound is finally given by:

$$\begin{aligned}
\mathbf{Rel}^{\mathrm{FF}}(\varnothing) &= \mathbb{E}_{x_{1:n}}\mathbb{E}_{\epsilon_{1:n}} \inf_{f\in\mathcal{F}}\left\{-\sum_{s=1}^{n} f(x_s)^T Y_s^{(0)}\right\} \\
&= \mathbb{E}_{x_{1:n}}\mathbb{E}_{\epsilon_{1:n}} - \sup_{f\in\mathcal{F}}\left\{\sum_{s=1}^{n} f(x_s)^T Y_s^{(0)}\right\} \\
&= \mathbb{E}_{x_{1:n}}\mathbb{E}_{\epsilon_{1:n}} - \sup_{f\in\mathcal{F}}\left\{\sum_{s=1}^{n} f(x_s)^T 2\epsilon_s\right\} \\
&= \mathbb{E}_{x_{1:n}} - 2\mathscr{R}(\mathcal{F}, x_{1:n}).
\end{aligned}$$

Thus, by Lemma 4.1, we have that

$$\mathbb{E}_{x_{1:n}}(\mathbf{Regret}) \leqslant \mathbb{E}_{x_{1:n}} 2\mathscr{R}(\mathcal{F}, x_{1:n}).$$

This completes the proof. ■

Proof of Proposition 4.1

As shown in Theorem 4.1, in order to bound $\mathbb{E}_{x_{1:n}}(\mathbf{Regret})$, we need to bound the quantity

$$\mathbb{E}_{x_{1:n},\epsilon_{1:n}}\left[\sup_{f\in\mathcal{F}}\sum_{s=1}^{n} f(x_s)^T\epsilon_s\right].$$

Denote by $z_f \coloneqq \sum_{s=1}^{n} f(x_s)^T\epsilon_s$. Then, for $\lambda > 0$, we have that

$$
\begin{aligned}
e^{\lambda \mathbb{E}_{x_{1:n},\epsilon_{1:n}}[\sup z_f]} &\overset{(a)}{\leqslant} \mathbb{E}_{x_{1:n},\epsilon_{1:n}}\big[e^{\lambda \sup z_f}\big] \\
&\overset{(b)}{=} \mathbb{E}_{x_{1:n},\epsilon_{1:n}}\big[\sup e^{\lambda z_f}\big] \\
&\overset{(c)}{\leqslant} \sum_{f \in \mathcal{F}} \mathbb{E}_{x_{1:n},\epsilon_{1:n}}\big[e^{\lambda z_f}\big] \\
&= \sum_{f \in \mathcal{F}} \mathbb{E}_{x_{1:n},\epsilon_{1:n}} \left[\prod_{s=1}^{n} e^{\lambda f(x_s)^T \epsilon_s} \right] \\
&\overset{(d)}{=} \sum_{f \in \mathcal{F}} \prod_{s=1}^{n} \mathbb{E}_{x_s,\epsilon_s}\big[e^{\lambda f(x_s)^T \epsilon_s}\big],
\end{aligned}
$$

where (a) follows from Jensen's inequality, (b) follows from the monotonicity of the exponential function, (c) follows from the fact that $|\mathcal{F}| < \infty$, and (d) follows from independence of the contexts and the Rademacher random variables (across time). Moreover, note that, since $\mathbb{E}[f(x_s)^T \epsilon_s] = 0$ and $\mathbb{E}[(f(x_s)^T \epsilon_s)^k] \leqslant \mathrm{Var}[f(x_s)^T \epsilon_s] d^{k-2} = \mathrm{Var}(f) d^{k-2}$ for $k \geqslant 2$, we have that

$$
\mathbb{E}_{x_s,\epsilon_s}\big[e^{\lambda f(x_s)^T \epsilon_s}\big] = 1 + \sum_{k=2}^{\infty} \frac{\lambda^k \mathbb{E}[(f(x_s)^T \epsilon_s)^k]}{k!} \leqslant 1 + \frac{\mathrm{Var}(f)}{d^2}(e^{\lambda d} - \lambda d - 1).
$$

Therefore, letting $\lambda = \mu/\sqrt{n}$ for some $\mu > 0$ (to be chosen later), the overall bound is

$$\begin{aligned}
\mathbb{E}_{x_{1:n},\epsilon_{1:n}}[\sup z_f] &\leqslant \frac{\sqrt{n}}{\mu}\log\left[\sum_{f\in\mathcal{F}}\left(1+\frac{\mathrm{Var}(f)}{d^2}(e^{\lambda d}-\lambda d-1)\right)^n\right] \\
&\overset{(a)}{\leqslant} \frac{\sqrt{n}}{\mu}\log\left[\sum_{f\in\mathcal{F}}\left(1+\frac{\mathrm{Var}(f)}{d^2}(\lambda d)^2\right)^n\right] \\
&= \frac{\sqrt{n}}{\mu}\log\left[\sum_{f\in\mathcal{F}}\left(1+\frac{\mathrm{Var}(f)\mu^2}{n}\right)^n\right] \\
&\overset{(b)}{\leqslant} \frac{\sqrt{n}}{\mu}\log\left[\sum_{f\in\mathcal{F}}e^{\mathrm{Var}(f)\mu^2}\right] \\
&\leqslant \frac{\sqrt{n}}{\mu}\log\left[|\mathcal{F}|e^{\mathrm{Var}(f^*)\mu^2}\right],
\end{aligned}$$

where (a) follows from the inequality $e^x - x - 1 \leqslant x^2$ for $x < 1.79$, and (b) follows from the inequality $(1 + a/n)^n \leqslant e^a$ for $a \geqslant 0$. Now, minimizing the right hand side with respect to μ, we obtain $\mu^* = \sqrt{\frac{\log(|\mathcal{F}|)}{\mathrm{Var}(f^*)}}$. Hence

$$\mathbb{E}_{x_{1:n},\epsilon_{1:n}}[\sup z_f] \leqslant 2\sqrt{n\log|\mathcal{F}|\mathrm{Var}(f^*)}.$$

Note that in order for this bound to hold, we must have (according to (a)) that

$$\lambda^* d < 1.79 \iff \sqrt{\frac{\log(|\mathcal{F}|)}{n\mathrm{Var}(f^*)}}d < 1.79 \iff n > \frac{d^2\log(|\mathcal{F}|)}{(1.79^2)\mathrm{Var}(f^*)} \geqslant \frac{d\log(|\mathcal{F}|)}{(1.79^2)}.$$

Given this, the final bound on the expected regret is then

$$\mathbb{E}_{x_{1:n}}(\textbf{Regret}) \leqslant 4\sqrt{n\log(|\mathcal{F}|)\mathrm{Var}(f^*)}.$$

This completes the proof. ■

Proof of Lemma 4.2

We prove this lemma by induction. For $n = 0$ the statement clearly holds true because both sides of equation (4.10) are zero. Now suppose the lemma holds for some $n \geqslant 0$.

Then, for δ Rademacher random variable

$$\mathbb{E}_{\epsilon_{1:n+1}}\left[\sup_M \sum_{s=1}^{n+1} \phi_s(M)^T \epsilon_s\right] = \mathbb{E}_{\epsilon_{1:n+1}}\left[\sup_M \sum_{s=1}^{n} \phi_s(M)^T \epsilon_s + \phi_{n+1}(M)^T \epsilon_{n+1}\right]$$

$$= \mathbb{E}_{\epsilon_{1:n+1}}\mathbb{E}_\delta\left[\sup_M \sum_{s=1}^{n} \phi_s(M)^T \epsilon_s + \phi_{n+1}(M)^T \epsilon_{n+1}\delta\right]$$

$$= \mathbb{E}_{\epsilon_{1:n+1}}\left\{\frac{1}{2}\left[\sup_M \sum_{s=1}^{n} \phi_s(M)^T \epsilon_s + \phi_{n+1}(M)^T \epsilon_{n+1}\right] + \right.$$

$$\left. + \frac{1}{2}\left[\sup_{M'} \sum_{s=1}^{n} \phi_s(M')^T \epsilon_s - \phi_{n+1}(M')^T \epsilon_{n+1}\right]\right\}$$

$$= \mathbb{E}_{\epsilon_{1:n+1}}\left[\sup_{M,M'} \sum_{s=1}^{n} \left(\frac{\phi_s(M) + \phi_s(M')}{2}\right)^T \epsilon_s + \left(\frac{\phi_{n+1}(M) - \phi_{n+1}(M')}{2}\right)^T \epsilon_{n+1}\right]$$

$$\overset{(a)}{=} \mathbb{E}_{\epsilon_{1:n}}\left[\sup_{M,M'} \sum_{s=1}^{n} \left(\frac{\phi_s(M) + \phi_s(M')}{2}\right)^T \epsilon_s + \frac{1}{2}\|\phi_{n+1}(M) - \phi_{n+1}(M')\|_1\right]$$

$$\overset{(b)}{\leqslant} \mathbb{E}_{\epsilon_{1:n}}\left[\sup_{M,M'} \sum_{s=1}^{n} \left(\frac{\phi_s(M) + \phi_s(M')}{2}\right)^T \epsilon_s + \frac{1}{2}\|\psi_{n+1}(M) - \psi_{n+1}(M')\|_1\right]$$

$$\overset{(c)}{=} \mathbb{E}_{\epsilon_{1:n+1}}\left[\sup_{M,M'} \sum_{s=1}^{n} \left(\frac{\phi_s(M) + \phi_s(M')}{2}\right)^T \epsilon_s + \left(\frac{\psi_{n+1}(M) - \psi_{n+1}(M')}{2}\right)^T \epsilon_{n+1}\right]$$

$$\overset{(d)}{=} \mathbb{E}_{\epsilon_{1:n+1}}\mathbb{E}_\delta\left[\sup_M \sum_{s=1}^{n} \phi_s(M)^T \epsilon_s + \psi_{n+1}(M)^T \epsilon_{n+1}\delta\right]$$

$$= \mathbb{E}_{\epsilon_{1:n+1}}\left[\sup_M \sum_{s=1}^{n} \phi_s(M)^T \epsilon_s + \psi_{n+1}(M)^T \epsilon_{n+1}\right]$$

$$\overset{(e)}{\leqslant} \mathbb{E}_{\epsilon_{1:n+1}}\left[\sup_M \sum_{s=1}^{n} \psi_s(M)^T \epsilon_s + \psi_{n+1}(M)^T \epsilon_{n+1}\right]$$

$$= \mathbb{E}_{\epsilon_{1:n+1}}\left[\sup_M \sum_{s=1}^{n+1} \psi_s(M)^T \epsilon_s\right],$$

where step (a) follows from the fact that it must be the case that whenever the j-th entry of ϵ_{n+1} is $+1$, the j-th entry of $\phi_{n+1}(M)$ must be larger that the j-th entry of $\phi_{n+1}(M')$ and whenever the j-th entry of ϵ_{n+1} is -1, the j-th entry of $\phi_{n+1}(M)$ must be smaller that the j-th entry of $\phi_{n+1}(M')$. This must hold because otherwise M and M' could not be the maximizers (because we can just swap the j-th entries and increase the objective). Step (b) follows from the assumption; step (c) holds for the same reason as step (a); step (d) follows from reversing the above steps (applied to ψ_{n+1}), and step (e) follows from the induction hypothesis. ■

Proof of Corollary 4.1

We apply Lemma 4.2 with $\mathcal{M} = \mathcal{V}$, $M = v$, $\phi_s(M) = \gamma(v(x_s))$ and $\psi_s(M) = Cv(x_s)$. Given the assumption on γ, we have that $\|\gamma(v(x_s)) - \gamma(v'(x_s))\|_1 \leqslant C\|v(x_s) - v'(x_s)\|_1$, i.e., the conditions of Lemma 4.2 hold, yielding equation (4.11). ■

Proof of Lemma 4.3

We want to bound the following quantity

$$\mathbb{E}_{\epsilon_{1:n}} \sup_{M \in \mathbb{R}^{d \times N}} \left[\sum_{s=1}^{n} (Mx_s)^T \epsilon_s \right].$$

Denoting by m_i the i-th column of M and by $[x_s]_i$ the i-th element of x_s, we can rewrite $(Mx_s)^T \epsilon_s$ as follows:

$$(Mx_s)^T \epsilon_s = \epsilon_s^T (Mx_s) = \epsilon_s^T \left(\sum_{i=1}^{N} [x_s]_i m_i \right) = \sum_{i=1}^{N} [x_s]_i \epsilon_s^T m_i.$$

Therefore,

$$\mathbb{E}_{\epsilon_{1:n}} \sup_{M\in\mathbb{R}^{d\times N}} \left[\sum_{s=1}^{n}(Mx_s)^T\epsilon_s\right] =$$

$$= \mathbb{E}_{\epsilon_{1:n}} \sup_{M\in\mathbb{R}^{d\times N}} \left[\sum_{s=1}^{n}\left(\sum_{i=1}^{N}[x_s]_i\epsilon_s^T m_i\right)\right]$$

$$= \mathbb{E}_{\epsilon_{1:n}} \sup_{M\in\mathbb{R}^{d\times N}} \left[\sum_{i=1}^{N}\left(\sum_{s=1}^{n}[x_s]_i\epsilon_s^T m_i\right)\right]$$

$$= \mathbb{E}_{\epsilon_{1:n}} \sup_{M\in\mathbb{R}^{d\times N}} \left[\sum_{i=1}^{N}\left(\sum_{s=1}^{n}[x_s]_i\epsilon_s\right)^T m_i\right]$$

$$\overset{(a)}{\leqslant} \mathbb{E}_{\epsilon_{1:n}} \sup_{M\in\mathbb{R}^{d\times N}} \left[\sum_{i=1}^{N}\left(\left\|\sum_{s=1}^{n}[x_s]_i\epsilon_s\right\|_2 \|m_i\|_2\right)\right]$$

$$\overset{(b)}{\leqslant} C\mathbb{E}_{\epsilon_{1:n}} \left(\sum_{i=1}^{N}\left\|\sum_{s=1}^{n}[x_s]_i\epsilon_s\right\|_2\right)$$

$$= C\sum_{i=1}^{N}\mathbb{E}_{\epsilon_{1:n}} \left(\left\|\sum_{s=1}^{n}[x_s]_i\epsilon_s\right\|_2\right),$$

where (a) follows from Cauchy-Schwarz and (b) follows from our hypothesis. Now, denoting by $[w_i]_l = \sum_{s=1}^{n}[x_s]_i[\epsilon_s]_l$ the l-th element of the i-th vector $\sum_{s=1}^{n}[x_s]_i\epsilon_s$, we have that

$$\mathbb{E}_{\epsilon_{1:n}}\left(\left\|\sum_{s=1}^{n}[x_s]_i\epsilon_s\right\|_2\right) = \mathbb{E}_{\epsilon_{1:n}}\left(\sqrt{\sum_{l=1}^{d}([w_i]_l)^2}\right)$$

$$\overset{(a)}{\leqslant} \sqrt{\sum_{l=1}^{d}\mathbb{E}_{\epsilon_{1:n}}([w_i]_l)^2}$$

$$= \sqrt{d\sum_{s=1}^{n}([x_s]_i)^2}$$

$$\leqslant \sqrt{dn\max_{s,i}([x_s]_i)^2}.$$

where (a) follows from Jensen's inequality. Therefore, the final bound is given by

$$\mathbb{E}_{\epsilon_{1:n}}\sup_{M\in\mathbb{R}^{d\times N}}\left[\sum_{s=1}^{n}(Mx_s)^T\epsilon_s\right] \leqslant C\sum_{i=1}^{N}\mathbb{E}_{\epsilon_{1:n}}\left(\left\|\sum_{s=1}^{n}[x_s]_i\epsilon_s\right\|_2\right) \leqslant CN\sqrt{dn\max_{s,i}([x_s]_i)^2}.$$

This completes the proof. ■

Proof of Corollary 4.2

As shown in Theorem 4.1, the upper-bound on the expected regret is given by

$$\mathbb{E}_{x_{1:n}}2\mathscr{R}(\mathcal{F}, x_{1:n}).$$

Moreover, by using Proposition 4.2, we have that

$$\mathbb{E}_{x_{1:n}}(\textbf{Regret}) \leqslant 2\sqrt{dn}\left(CN\mathbb{E}_{x_{1:n}}\left[\sqrt{\max_{s,i}([x_s]_i)^2}\right] + d\sqrt{2\log(2)}\right).$$

Therefore, it is enough to bound $\mathbb{E}\left[\sqrt{\max_{s,i}([x_s]_i)^2}\right]$. In order to do that, note that for any $\mu > 0$, we have

$$\begin{aligned}
\mathbb{E}\left[\sqrt{\max_{s,i}([x_s]_i)^2}\right] &\overset{(a)}{=} \mathbb{E}\left[\max_{s,i}|[x_s]_i|\right] \overset{(b)}{=} \mathbb{E}\left[\max_{s,i}[x_s]_i\right] = \frac{1}{\mu}\mathbb{E}\left[\log e^{\mu \max_{s,i}[x_s]_i}\right] \\
&\overset{(c)}{\leqslant} \frac{1}{\mu}\log \mathbb{E}\left[e^{\mu \max_{s,i}[x_s]_i}\right] \\
&\overset{(d)}{=} \frac{1}{\mu}\log \mathbb{E}\left[\max_{s,i} e^{\mu [x_s]_i}\right] \\
&\leqslant \frac{1}{\mu}\log\left(\sum_{s,i}\mathbb{E}\left[e^{\mu [x_s]_i}\right]\right) \\
&\overset{(e)}{\leqslant} \frac{1}{\mu}\log\left(nNe^{\mu\eta+\mu^2/8}\right) \\
&= \frac{1}{\mu}\log(nN) + \frac{\mu}{8} + \eta,
\end{aligned}$$

where (a) holds because the square root function is monotone; (b) follows from the assumption that $[x_s]_i \in [0,1]$ for any s and i; (c) follows from Jensen's inequality applied to the log function; (d) holds because the exponential function is monotone and (e) follows from Hoeffding's lemma applied to the random variable $[x_s]_i$.

Minimizing the last expression with respect to μ, we obtain $\mu^* = \sqrt{8\log(nN)}$, leading to the final bound

$$\mathbb{E}_{x_{1:n}}(\textbf{Regret}) \leqslant 2\sqrt{dn}\left(CN\left(\sqrt{\frac{\log(nN)}{2}} + \eta\right) + d\sqrt{2\log(2)}\right).$$

This completes the proof. ■

Proof of Lemma 4.4

The optimization problem that we want to solve is the following:

$$\sup_{q\in\Delta_{\mathcal{A}}}\min_{v\in\mathcal{D}}\left\{q^T A v - \max_{f\in\mathcal{F}}\left\{\sum_{s\neq t}^{n} f(x_s)^T Y_s^{(t)} + f(x_t)^T v\right\}\right\}. \tag{4.36}$$

Denote by $\psi_i \coloneqq \max_{f\in\mathcal{F}} \left\{\sum_{s\neq t}^{n} f(x_s)^T Y_s^{(t)} + f(x_t)^T v_i\right\}$, where i is the index for the i-th element of the set $\mathcal{D} = \{-1, 1\}^d$. Note that each ψ_i can be computed with a single oracle access, for a total of 2^d accesses. Thus, we can assume that they are given. With this notation, the optimization problem in (4.36) can be rewritten as

$$\begin{aligned}\sup_{q\in\Delta_{\mathcal{A}}} \min_{i\in[2^d]} \left\{q^T A v_i - \psi_i\right\} = \sup_{q\in\Delta_{\mathcal{A}}} \quad & z \\ \text{subject to} \quad & z \leqslant q^T A v_i - \psi_i, \quad i = 1, \ldots, 2^d. \end{aligned} \tag{4.37}$$

Note that (4.37) is a linear program with 2^d many variables and 2^d many constraints. Thus, it can be solved in time $O(2^d)$. ■

Proof of Lemma 4.5

We prove this lemma similarly to Lemma 4.1. Let $(q_t)_{t=1}^n$ be an admissible strategy and define $\overline{\textbf{Regret}} = -\textbf{Regret}$. Then,

$$\begin{aligned}
\mathbb{E}_{x_{1:n}}(\overline{\textbf{Regret}}) &\geqslant \inf_{v_{1:n}} \mathbb{E}_{x_{1:n}}(\overline{\textbf{Regret}}) \\
&\overset{(a)}{=} \inf_{v_1}\inf_{v_2}\cdots\inf_{v_{n-1}}\inf_{v_n} \mathbb{E}_{x_{1:n}}(\overline{\textbf{Regret}}) \\
&\overset{(b)}{\geqslant} \inf_{v_1}\inf_{v_2}\cdots\inf_{v_{n-1}} \mathbb{E}_{x_{1:n}}(\inf_{v_n}\overline{\textbf{Regret}}) \\
&\overset{(c)}{=} \inf_{v_1}\inf_{v_2}\cdots\inf_{v_{n-1}} \mathbb{E}_{x_{1:(n-1)}}\mathbb{E}_{x_n}(\inf_{v_n}\overline{\textbf{Regret}}) \\
&\geqslant \inf_{v_1}\inf_{v_2}\cdots\mathbb{E}_{x_{1:(n-1)}}\inf_{v_{n-1}}\mathbb{E}_{x_n}(\inf_{v_n}\overline{\textbf{Regret}}) \\
&\;\;\vdots \\
&\geqslant \mathbb{E}_{x_1}\inf_{v_1}\mathbb{E}_{x_2}\inf_{v_2}\cdots\mathbb{E}_{x_{n-1}}\inf_{v_{n-1}}\mathbb{E}_{x_n}(\inf_{v_n}\overline{\textbf{Regret}}),
\end{aligned}$$

where (a) follows from the fact that, for bounded f, $\inf_{(x,y)} f(x,y) = \inf_x \inf_y f(x,y)$; (b) follows from $\inf_{y\in\mathcal{Y}} \mathbb{E}[f(X,y)] \geqslant \mathbb{E}[\inf_{y\in\mathcal{Y}} f(X,y)]$ for any measurable f; and (c)

follows from the independence of the contexts. Now, by admissibility of $\mathbf{Rel}^{\mathrm{BF}}(\cdot)$, we have that for all $x_{1:n}$, $\tilde{v}_{1:n}$ and $q_{1:n}$,

$$\mathbb{E}_{\hat{a}_{1:n}\sim q_{1:n},X_{1:n}}\mathbf{Rel}^{\mathrm{BF}}(I_{1:n}) \leqslant -\sup_{f\in\mathcal{F}}\sum_{t=1}^{n} e_{f(x_t)}^{T}\tilde{v}_t.$$

Thus, using the definition of regret (in the context of bandit feedback), we can continue the lower bound as follows:

$$\mathbb{E}_{x_{1:n}}(\overline{\mathbf{Regret}})$$

$$\geqslant \mathbb{E}_{x_1} \inf_{\tilde{v}_1} \mathbb{E}_{x_2} \inf_{\tilde{v}_2} \cdots \mathbb{E}_{x_{n-1}} \inf_{\tilde{v}_{n-1}} \mathbb{E}_{x_n} \inf_{\tilde{v}_n} \left(\sum_{t=1}^{n} q_t^T \tilde{v}_t - \sup_{f \in \mathcal{F}} \sum_{t=1}^{n} e_{f(x_t)}^T \tilde{v}_t \right)$$

$$\overset{(a)}{\geqslant} \mathbb{E}_{x_1} \inf_{\tilde{v}_1} \mathbb{E}_{x_2} \inf_{\tilde{v}_2} \cdots \mathbb{E}_{x_{n-1}} \inf_{\tilde{v}_{n-1}} \mathbb{E}_{x_n} \inf_{\tilde{v}_n} \left(\sum_{t=1}^{n} q_t^T \tilde{v}_t + \mathbb{E}_{\hat{a}_{1:n} \sim q_{1:n}, X_{1:n}} \mathbf{Rel}^{\mathrm{BF}}(I_{1:n}) \right)$$

$$\overset{(b)}{=} \mathbb{E}_{x_1} \inf_{\tilde{v}_1} \mathbb{E}_{x_2} \inf_{\tilde{v}_2} \cdots \mathbb{E}_{x_{n-1}} \inf_{\tilde{v}_{n-1}} \mathbb{E}_{x_n} \inf_{\tilde{v}_n} \left(\mathbb{E}_{\hat{a}_{1:n} \sim q_{1:n}, X_{1:n}} \left[\sum_{t=1}^{n} \tilde{v}_t(\hat{a}_t) + \mathbf{Rel}^{\mathrm{BF}}(I_{1:n}) \right] \right)$$

$$= \mathbb{E}_{x_1} \inf_{\tilde{v}_1} \mathbb{E}_{x_2} \inf_{\tilde{v}_2} \cdots \mathbb{E}_{x_{n-1}} \inf_{\tilde{v}_{n-1}} \mathbb{E}_{x_n} \inf_{\tilde{v}_n} \mathbb{E}_{\hat{a}_{1:(n-1)}, X_{1:(n-1)}} \mathbb{E}_{\hat{a}_n, X_n} \left[\sum_{t=1}^{n} \tilde{v}_t(\hat{a}_t) + \mathbf{Rel}^{\mathrm{BF}}(I_{1:n}) \right]$$

$$\geqslant \mathbb{E}_{x_1} \inf_{\tilde{v}_1} \mathbb{E}_{x_2} \inf_{\tilde{v}_2} \cdots \mathbb{E}_{x_{n-1}} \inf_{\tilde{v}_{n-1}} \mathbb{E}_{x_n} \mathbb{E}_{\hat{a}_{1:(n-1)}, X_{1:(n-1)}} \inf_{\tilde{v}_n} \mathbb{E}_{\hat{a}_n, X_n} \left[\sum_{t=1}^{n} \tilde{v}_t(\hat{a}_t) + \mathbf{Rel}^{\mathrm{BF}}(I_{1:n}) \right]$$

$$\overset{(c)}{=} \mathbb{E}_{x_1} \inf_{\tilde{v}_1} \mathbb{E}_{x_2} \inf_{\tilde{v}_2} \cdots \mathbb{E}_{x_{n-1}} \inf_{\tilde{v}_{n-1}} \mathbb{E}_{\hat{a}_{1:(n-1)}, X_{1:(n-1)}} \mathbb{E}_{x_n} \inf_{\tilde{v}_n} \mathbb{E}_{\hat{a}_n, X_n} \left[\sum_{t=1}^{n} \tilde{v}_t(\hat{a}_t) + \mathbf{Rel}^{\mathrm{BF}}(I_{1:n}) \right]$$

$$\vdots$$

$$\overset{(d)}{\geqslant} \mathbb{E}_{x_1} \inf_{\tilde{v}_1} \mathbb{E}_{\hat{a}_1 \sim q_1, X_1} \mathbb{E}_{x_2} \inf_{\tilde{v}_2} \mathbb{E}_{\hat{a}_2 \sim q_2, X_2} \cdots \mathbb{E}_{x_n} \inf_{\tilde{v}_n} \mathbb{E}_{\hat{a}_n \sim q_n, X_n} \left[\sum_{t=1}^{n} \tilde{v}_t(\hat{a}_t) + \mathbf{Rel}^{\mathrm{BF}}(I_{1:n}) \right]$$

$$= \mathbb{E}_{x_1} \inf_{\tilde{v}_1} \mathbb{E}_{\hat{a}_1 \sim q_1, X_1} \cdots \mathbb{E}_{x_n} \inf_{\tilde{v}_n} \mathbb{E}_{\hat{a}_n \sim q_n, X_n} \left[\sum_{t=1}^{n-1} \tilde{v}_t(\hat{a}_t) + \tilde{v}_n(\hat{a}_n) + \mathbf{Rel}^{\mathrm{BF}}(I_{1:n}) \right]$$

$$= \mathbb{E}_{x_1} \inf_{\tilde{v}_1} \mathbb{E}_{\hat{a}_1 \sim q_1, X_1} \cdots \mathbb{E}_{x_{n-1}} \inf_{\tilde{v}_{n-1}} \mathbb{E}_{\hat{a}_{n-1} \sim q_{n-1}, X_{n-1}} \left[\sum_{t=1}^{n-1} \tilde{v}_t(\hat{a}_t) + \right.$$

$$\left. + \mathbb{E}_{x_n} \inf_{\tilde{v}_n} \mathbb{E}_{\hat{a}_n, X_n} \left\{ \tilde{v}_n(\hat{a}_n) + \mathbf{Rel}^{\mathrm{BF}}(I_{1:n}) \right\} \right]$$

$$\overset{(e)}{\geqslant} \mathbb{E}_{x_1} \inf_{\tilde{v}_1} \mathbb{E}_{\hat{a}_1 \sim q_1, X_1} \cdots \mathbb{E}_{x_{n-1}} \inf_{\tilde{v}_{n-1}} \mathbb{E}_{\hat{a}_{n-1} \sim q_{n-1}, X_{n-1}} \left[\sum_{t=1}^{n-1} \tilde{v}_t(\hat{a}_t) + \mathbf{Rel}^{\mathrm{BF}}(I_{1:n-1}) \right],$$

where (a) holds by admissibility; (b) follows from the fact that $\mathbb{E}_{\hat{a}_t \sim q_t, X_t}\left[\tilde{v}_t(\hat{a}_t)\right] = \mathbb{E}_{\hat{a}_t}\left[\hat{a}_t^T v_t\right] = q_t^T A v_t = q_t^T \tilde{v}_t$; (c) holds by Fubini Theorem and (e) holds by equation (4.17), since $(q_t)_{t=1}^n$ is admissible. Finally, proceeding iteratively from step (d), we obtain the final bound

$$\mathbb{E}_{x_{1:n}}(\overline{\mathbf{Regret}}) \geqslant \mathbf{Rel}^{\mathrm{BF}}(\varnothing),$$

i.e. $\mathbb{E}_{x_{1:n}}(\mathbf{Regret}) \leqslant -\mathbf{Rel}^{\mathrm{BF}}(\varnothing)$. This completes the proof. ■

Proof of Theorem 4.2

Admissibility: initial condition. For any $x_{1:n}, \tilde{v}_{1:n}$ and $q_{1:n}$ it holds that

$$-\sup_{f\in\mathcal{F}} \sum_{t=1}^n e_{f(x_t)}^T \tilde{v}_t = \inf_{f\in\mathcal{F}} -\sum_{t=1}^n e_{f(x_t)}^T \mathbb{E}_{\hat{a}_t, X_t}[\hat{v}_t] \geqslant \mathbb{E}_{\hat{a}_{1:n}, X_{1:n}} \inf_{f\in\mathcal{F}} -\sum_{t=1}^n e_{f(x_t)}^T \hat{v}_t$$
$$= \mathbb{E}_{\hat{a}_{1:n}, X_{1:n}} \mathbf{Rel}^{\mathrm{BF}}(I_{1:n}).$$

Admissibility: recursion. Let $\mathcal{L} = \{Le_i : i \in [2^d]\} \cup \{\mathbf{0}\}$ and let $\epsilon_s \in \{-1, +1\}^{2^d}$ denote a vector of independent Rademacher random variables. We will now reason conditionally on x_t. Let us denote by

$$\rho_t := (x_{(t+1):n}, \epsilon_{(t+1):n}, Z_{(t+1):n}),$$

a draw of independent covariates from $\mathbb{P}_x$, independent Rademacher random variables and real-valued i.i.d. random variables for the "future rounds". Define the randomized prediction algorithm as

$$q_t = \mathbb{E}_{\rho_t}[q_t(\rho_t)] \quad \text{with} \quad q_t(\rho_t) = \left(1 - \frac{2^d}{L}\right) q_t^*(\rho_t) + \frac{1}{L}\mathbf{1}, \tag{4.38}$$

and

$$q_t^* = \mathbb{E}_{\rho_t}[q_t^*(\rho_t)] \quad \text{with} \quad q_t^*(\rho_t) = \mathrm{argmax}_{q\in\Delta_{\mathcal{A}}} \inf_{p_t \in \Delta'_{\mathcal{L}}} \mathbb{E}_{\hat{v}_t \sim p_t}\left[q^T \hat{v}_t + R(x_{1:t}, \hat{v}_{1:t}, \rho_t)\right], \tag{4.39}$$

where $\Delta_{\mathcal{A}}$ denotes the set of distributions over $\mathcal{A}$ and $\Delta'_{\mathcal{L}} = \{p \in \Delta_{\mathcal{L}} : p(i) \leqslant d/L \ \forall i \in [2^d]\}$. First note that

$$\mathbb{E}_{\hat{a}_t \sim q_t, X_t}\left[\tilde{v}_t(\hat{a}_t)\right] \overset{(a)}{=} q_t^T \tilde{v}_t = \left(1 - \frac{2^d}{L}\right) q_t^{*T} \tilde{v}_t + \frac{1}{L} \mathbf{1}^T \tilde{v}_t \tag{4.40}$$

$$\overset{(b)}{\geqslant} q_t^{*T} \tilde{v}_t - \frac{2^d}{L} q_t^{*T}(d\mathbf{1}) - \frac{d2^d}{L} \tag{4.41}$$

$$= \mathbb{E}_{\hat{a}_t, X_t}\left[q_t^{*T} \hat{v}_t\right] - \frac{d2^{d+1}}{L}, \tag{4.42}$$

where (a) follows from the fact that $\mathbb{E}_{\hat{a}_t \sim q_t, X_t}\left[\tilde{v}_t(\hat{a}_t)\right] = \mathbb{E}_{\hat{a}_t}\left[\hat{a}_t^T v_t\right] = q_t^T A v_t = q_t^T \tilde{v}_t$ and (b) holds because $\tilde{v}_t \in [-d, d]^{2^d}$. Thus, we have that

$$\begin{aligned}
&\min_{\tilde{v}_t} \mathbb{E}_{\hat{a}_t \sim q_t, X_t}\left\{\tilde{v}_t(\hat{a}_t) + \mathbf{Rel}^{\mathrm{BF}}(I_{1:t})\right\} \\
&\overset{(a)}{\geqslant} \min_{\tilde{v}_t} \mathbb{E}_{\hat{a}_t \sim q_t, X_t}\left\{q_t^{*T} \hat{v}_t + \mathbf{Rel}^{\mathrm{BF}}(I_{1:t})\right\} - \frac{d2^{d+1}}{L} \\
&= \min_{\tilde{v}_t} \mathbb{E}_{\hat{a}_t \sim q_t, X_t}\left\{q_t^{*T} \hat{v}_t + \mathbb{E}_{\rho_t}\left[R(x_{1:t}, \hat{v}_{1:t}, \rho_t)\right]\right\} - \frac{d2^{d+1}}{L} \\
&= \min_{\tilde{v}_t} \mathbb{E}_{\hat{a}_t \sim q_t, X_t}\left\{\mathbb{E}_{\rho_t}\left[q_t^*(\rho_t)^T \hat{v}_t + R(x_{1:t}, \hat{v}_{1:t}, \rho_t)\right]\right\} - \frac{d2^{d+1}}{L} \\
&\overset{(b)}{\geqslant} \inf_{\hat{v}_t}\left\{\mathbb{E}_{\rho_t}\left[q_t^*(\rho_t)^T \hat{v}_t + R(x_{1:t}, \hat{v}_{1:t}, \rho_t)\right]\right\} - \frac{d2^{d+1}}{L},
\end{aligned}$$

where (a) follows from equation (4.42) and (b) follows from the fact that we are taking the infimum over a larger set (since $\hat{v}_t$ is a function of $\tilde{v}_t, \hat{a}_t$ and X_t). Observe now that $\hat{v}_t$ is a random variable taking values in $\mathcal{L}$ and such that the probability that is equal to Le_i can be upper bounded as:

$$\mathbb{P}(\hat{v}_t = Le_i) = \mathbb{E}_{\rho_t}[\mathbb{P}(\hat{v}_t = Le_i \mid \rho_t)] = \mathbb{E}_{\rho_t}\left[q_t(\rho_t)(i) \frac{\tilde{v}_t(i)}{L q_t(\rho_t)(i)}\right] \leqslant d/L.$$

Therefore, we can continue the lower bound as follows:

$$\min_{\tilde{v}_t} \mathbb{E}_{\hat{a}_t \sim q_t, X_t} \left\{ \tilde{v}_t(\hat{a}_t) + \mathbf{Rel}^{\mathrm{BF}}(I_{1:t}) \right\}$$

$$\overset{(a)}{\geqslant} \inf_{p_t \in \Delta'_{\mathcal{L}}} \mathbb{E}_{\hat{v}_t \sim p_t} \left\{ \mathbb{E}_{\rho_t} \left[q_t^*(\rho_t)^T \hat{v}_t + R(x_{1:t}, \hat{v}_{1:t}, \rho_t) \right] \right\} - \frac{d 2^{d+1}}{L}$$

$$\overset{(b)}{\geqslant} \mathbb{E}_{\rho_t} \inf_{p_t \in \Delta'_{\mathcal{L}}} \mathbb{E}_{\hat{v}_t \sim p_t} \left\{ q_t^*(\rho_t)^T \hat{v}_t + R(x_{1:t}, \hat{v}_{1:t}, \rho_t) \right\} - \frac{d 2^{d+1}}{L}$$

$$\overset{(c)}{=} \mathbb{E}_{\rho_t} \sup_{q \in \Delta_{\mathcal{A}}} \inf_{p_t \in \Delta'_{\mathcal{L}}} \mathbb{E}_{\hat{v}_t \sim p_t} \left\{ q^T \hat{v}_t + R(x_{1:t}, \hat{v}_{1:t}, \rho_t) \right\} - \frac{d 2^{d+1}}{L}$$

$$\overset{(d)}{=} \mathbb{E}_{\rho_t} \inf_{p_t \in \Delta'_{\mathcal{L}}} \sup_{q \in \Delta_{\mathcal{A}}} \mathbb{E}_{\hat{v}_t \sim p_t} \left\{ q^T \hat{v}_t + R(x_{1:t}, \hat{v}_{1:t}, \rho_t) \right\} - \frac{d 2^{d+1}}{L}$$

$$= \mathbb{E}_{\rho_t} \inf_{p_t \in \Delta'_{\mathcal{L}}} \sup_{q \in \Delta_{\mathcal{A}}} \left\{ q^T \mathbb{E}_{\hat{v}_t \sim p_t}[\hat{v}_t] + \mathbb{E}_{\hat{v}_t \sim p_t} R(x_{1:t}, \hat{v}_{1:t}, \rho_t) \right\} - \frac{d 2^{d+1}}{L}$$

$$= \mathbb{E}_{\rho_t} \inf_{p_t \in \Delta'_{\mathcal{L}}} \left\{ \max_{j \in [2^d]} e_j^T \mathbb{E}_{\hat{v}_t \sim p_t}[\hat{v}_t] + \mathbb{E}_{\hat{v}_t \sim p_t} R(x_{1:t}, \hat{v}_{1:t}, \rho_t) \right\} - \frac{d 2^{d+1}}{L}$$

$$\overset{(e)}{=} \mathbb{E}_{\rho_t} \inf_{p_t \in \Delta'_{\mathcal{L}}} \left\{ \max_{j \in [2^d]} e_j^T \mathbb{E}_{\hat{v}_t \sim p_t}[\hat{v}_t] + \mathbb{E}_{\hat{v}_t \sim p_t} \inf_{f \in \mathcal{F}} \left\{ - \sum_{s=1}^{t} e_{f(x_s)}^T \hat{v}_s - \sum_{s=t+1}^{n} 2 e_{f(x_s)}^T \epsilon_s Z_s \right\} \right\} +$$

$$- \frac{(n-(t-1)) d 2^{d+1}}{L}$$

$$\overset{(f)}{=} \mathbb{E}_{\rho_t} \inf_{p_t \in \Delta'_{\mathcal{L}}} \left\{ \mathbb{E}_{\hat{v}_t \sim p_t} \inf_{f \in \mathcal{F}} \left\{ - \sum_{s=1}^{t} e_{f(x_s)}^T \hat{v}_s + \max_{j \in [2^d]} e_j^T \mathbb{E}_{\hat{v}_t \sim p_t}[\hat{v}_t] - \sum_{s=t+1}^{n} 2 e_{f(x_s)}^T \epsilon_s Z_s \right\} \right\}$$

$$- \frac{(n-(t-1)) d 2^{d+1}}{L},$$

where (a) follows from the fact that we are taking the infimum over all distributions p_t in $\Delta'_{\mathcal{L}}$; (b) follows from, first exchanging the expectations, and then using the fact that $\inf_{y \in \mathcal{Y}} \mathbb{E}[f(X,y)] \geqslant \mathbb{E}[\inf_{y \in \mathcal{Y}} f(X,y)]$; (c) follows from the definition of $q_t^*(\rho_t)$; (d) follows from the Minimax Theorem (since the objective is linear in both

q and p_t); (e) follows from the definition of $R(x_{1:t}, \hat{v}_{1:t}, \rho_t)$ and (f) holds because $\max_{j\in[2^d]} e_j^T \mathbb{E}_{\hat{v}_t\sim p_t}[\hat{v}_t]$ is a constant. Let us now focus on the first term and use the notation

$$S_{f,-t} := -\sum_{s=1}^{t-1} e_{f(x_s)}^T \hat{v}_s - \sum_{s=t+1}^{n} 2e_{f(x_s)}^T \epsilon_s Z_s.$$

The rest of the lower bounds will be derived conditionally on ρ_t. In particular, for δ_t one-dimensional Rademacher random variable, we have

$$\inf_{p_t\in\Delta'_{\mathcal{L}}} \left\{ \mathbb{E}_{\hat{v}_t\sim p_t} \inf_{f\in\mathcal{F}} \left\{ S_{f,-t} + \max_{j\in[2^d]} e_j^T \mathbb{E}_{\hat{v}_t\sim p_t}[\hat{v}_t] - e_{f(x_t)}^T \hat{v}_t \right\} \right\}$$

$$\overset{(a)}{\geqslant} \inf_{p_t\in\Delta'_{\mathcal{L}}} \left\{ \mathbb{E}_{\hat{v}_t\sim p_t} \inf_{f\in\mathcal{F}} \left\{ S_{f,-t} + e_{f(x_t)}^T \mathbb{E}_{\hat{v}_t\sim p_t}[\hat{v}_t] - e_{f(x_t)}^T \hat{v}_t \right\} \right\}$$

$$\overset{(b)}{\geqslant} \inf_{p_t\in\Delta'_{\mathcal{L}}} \left\{ \mathbb{E}_{\hat{v}_t\sim p_t, \hat{v}'_t\sim p_t} \inf_{f\in\mathcal{F}} \left\{ S_{f,-t} + e_{f(x_t)}^T (\hat{v}'_t - \hat{v}_t) \right\} \right\}$$

$$\overset{(c)}{=} \inf_{p_t\in\Delta'_{\mathcal{L}}} \left\{ \mathbb{E}_{\hat{v}_t\sim p_t, \hat{v}'_t\sim p_t} \mathbb{E}_{\delta_t} \inf_{f\in\mathcal{F}} \left\{ S_{f,-t} + \delta_t e_{f(x_t)}^T (\hat{v}'_t - \hat{v}_t) \right\} \right\}$$

where (a) follows from the fact that $\max_{j\in[2^d]} e_j^T \mathbb{E}_{\hat{v}_t\sim p_t}[\hat{v}_t] \geqslant e_{f(x_t)}^T \mathbb{E}_{\hat{v}_t\sim p_t}[\hat{v}_t]$; (b) follows from $\inf_{y\in\mathcal{Y}} \mathbb{E}[f(X,y)] \geqslant \mathbb{E}[\inf_{y\in\mathcal{Y}} f(X,y)]$ and (c) holds because $\mathbb{E}[g(X - X')] = \mathbb{E}[g(X' - X)]$, for any X, X' i.i.d. Now notice that the expression inside the first infimum (in the latter expression) can be decomposed as follows:

$$\mathbb{E}_{\hat{v}_t\sim p_t,\hat{v}'_t\sim p_t}\mathbb{E}_{\delta_t}\inf_{f\in\mathcal{F}}\left\{S_{f,-t}+\delta_t e^T_{f(x_t)}(\hat{v}'_t-\hat{v}_t)\right\}$$

$$=\mathbb{E}_{\hat{v}_t\sim p_t,\hat{v}'_t\sim p_t}\mathbb{E}_{\delta_t}\inf_{f\in\mathcal{F}}\left\{\frac{1}{2}S_{f,-t}+\delta_t e^T_{f(x_t)}\hat{v}'_t+\frac{1}{2}S_{f,-t}-\delta_t e^T_{f(x_t)}\hat{v}_t\right\}$$

$$\geqslant\mathbb{E}_{\hat{v}_t\sim p_t,\hat{v}'_t\sim p_t}\mathbb{E}_{\delta_t}\left\{\inf_{f\in\mathcal{F}}\left\{\frac{1}{2}S_{f,-t}+\delta_t e^T_{f(x_t)}\hat{v}'_t\right\}+\inf_{f\in\mathcal{F}}\left\{\frac{1}{2}S_{f,-t}-\delta_t e^T_{f(x_t)}\hat{v}_t\right\}\right\}$$

$$=\mathbb{E}_{\hat{v}_t\sim p_t,\hat{v}'_t\sim p_t}\left\{\mathbb{E}_{\delta_t}\inf_{f\in\mathcal{F}}\left\{\frac{1}{2}S_{f,-t}+\delta_t e^T_{f(x_t)}\hat{v}'_t\right\}+\mathbb{E}_{\delta_t}\inf_{f\in\mathcal{F}}\left\{\frac{1}{2}S_{f,-t}-\delta_t e^T_{f(x_t)}\hat{v}_t\right\}\right\}$$

$$=\mathbb{E}_{\hat{v}'_t\sim p_t}\mathbb{E}_{\delta_t}\inf_{f\in\mathcal{F}}\left\{\frac{1}{2}S_{f,-t}+\delta_t e^T_{f(x_t)}\hat{v}'_t\right\}+\mathbb{E}_{\hat{v}_t\sim p_t}\mathbb{E}_{\delta_t}\inf_{f\in\mathcal{F}}\left\{\frac{1}{2}S_{f,-t}-\delta_t e^T_{f(x_t)}\hat{v}_t\right\}$$

$$\overset{(a)}{=}\mathbb{E}_{\hat{v}'_t\sim p_t}\mathbb{E}_{\delta_t}\inf_{f\in\mathcal{F}}\left\{\frac{1}{2}S_{f,-t}+\delta_t e^T_{f(x_t)}\hat{v}'_t\right\}+\mathbb{E}_{\hat{v}_t\sim p_t}\mathbb{E}_{\delta_t}\inf_{f\in\mathcal{F}}\left\{\frac{1}{2}S_{f,-t}+\delta_t e^T_{f(x_t)}\hat{v}_t\right\}$$

$$=2\mathbb{E}_{\hat{v}_t\sim p_t}\mathbb{E}_{\delta_t}\inf_{f\in\mathcal{F}}\left\{\frac{1}{2}S_{f,-t}+\delta_t e^T_{f(x_t)}\hat{v}_t\right\}$$

$$=\mathbb{E}_{\hat{v}_t\sim p_t}\mathbb{E}_{\delta_t}\inf_{f\in\mathcal{F}}\left\{S_{f,-t}+2\delta_t e^T_{f(x_t)}\hat{v}_t\right\}$$

where (a) holds because δ_t and $-\delta_t$ have the same distribution. Thus,

$$\inf_{p_t\in\Delta'_{\mathcal{L}}}\left\{\mathbb{E}_{\hat{v}_t\sim p_t}\inf_{f\in\mathcal{F}}\left\{S_{f,-t}+\max_{j\in[2^d]}e_j^T\mathbb{E}_{\hat{v}_t\sim p_t}[\hat{v}_t]-e^T_{f(x_t)}\hat{v}_t\right\}\right\}$$

$$\geqslant\inf_{p_t\in\Delta'_{\mathcal{L}}}\left\{\mathbb{E}_{\hat{v}_t\sim p_t}\mathbb{E}_{\delta_t}\inf_{f\in\mathcal{F}}\left\{S_{f,-t}+2\delta_t e^T_{f(x_t)}\hat{v}_t\right\}\right\},$$

and so the overall bound is now given by

$$\min_{\tilde{v}_t}\mathbb{E}_{\hat{a}_t\sim q_t,X_t}\left\{\tilde{v}_t(\hat{a}_t)+\mathbf{Rel}^{\mathrm{BF}}(I_{1:t})\right\} \tag{4.43}$$

$$\geqslant\mathbb{E}_{\rho_t}\inf_{p_t\in\Delta'_{\mathcal{L}}}\left\{\mathbb{E}_{\hat{v}_t\sim p_t}\inf_{f\in\mathcal{F}}\left\{S_{f,-t}+\max_{j\in[2^d]}e_j^T\mathbb{E}_{\hat{v}_t\sim p_t}[\hat{v}_t]-e^T_{f(x_t)}\hat{v}_t\right\}\right\}-\frac{(n-(t-1))d2^{d+1}}{L} \tag{4.44}$$

$$\geqslant\mathbb{E}_{\rho_t}\inf_{p_t\in\Delta'_{\mathcal{L}}}\left\{\mathbb{E}_{\hat{v}_t\sim p_t}\mathbb{E}_{\delta_t}\inf_{f\in\mathcal{F}}\left\{S_{f,-t}+2\delta_t e^T_{f(x_t)}\hat{v}_t\right\}\right\}-\frac{(n-(t-1))d2^{d+1}}{L}. \tag{4.45}$$

Conditioning on $\hat{v}_t$, consider now a random variable M_t which is $\pm \max_i \hat{v}_t(i)$ with equal probability on the coordinates where $\hat{v}_t$ is equal to zero and equal to $\hat{v}_t$ on the coordinate that achieves the maximum. Clearly, $\mathbb{E}[M_t \mid \hat{v}_t] = \hat{v}_t$, i.e., M_t is an unbiased estimate of $\hat{v}_t$. Therefore,

$$\inf_{p_t \in \Delta'_{\mathcal{L}}} \left\{ \mathbb{E}_{\hat{v}_t \sim p_t} \mathbb{E}_{\delta_t} \inf_{f \in \mathcal{F}} \left\{ S_{f,-t} + 2\delta_t e^T_{f(x_t)} \hat{v}_t \right\} \right\} \tag{4.46}$$

$$= \inf_{p_t \in \Delta'_{\mathcal{L}}} \left\{ \mathbb{E}_{\hat{v}_t \sim p_t} \mathbb{E}_{\delta_t} \inf_{f \in \mathcal{F}} \left\{ S_{f,-t} + 2\delta_t e^T_{f(x_t)} \mathbb{E}[M_t \mid \hat{v}_t] \right\} \right\} \tag{4.47}$$

$$\geqslant \inf_{p_t \in \Delta'_{\mathcal{L}}} \left\{ \mathbb{E}_{\hat{v}_t \sim p_t} \mathbb{E}_{\delta_t} \mathbb{E}_{M_t} \inf_{f \in \mathcal{F}} \left\{ S_{f,-t} + 2\delta_t e^T_{f(x_t)} M_t \right\} \right\}. \tag{4.48}$$

Note that the random variable $\delta_t M_t$, conditioning on $\hat{v}_t$, is $\pm \max_i \hat{v}_t(i)$ with equal probability independently on each coordinate. Moreover, given any distribution $p_t \in \Delta'_{\mathcal{L}}$, the distribution of the maximum coordinate of $\hat{v}_t$ has support on $\{0, L\}$ and is equal to L with probability at most $d2^d/L$. Since the objective in (4.48) only depends on the distribution of the maximum coordinate of $\hat{v}_t$, we can continue the lower bound with an infimum over any distribution of random vectors whose coordinates are 0 with probability at least $1 - d2^d/L$ and otherwise $\pm L$ with equal probability. In particular, let ϵ_t be a 2^d-dimensional Rademacher random vector and denote by $a := \mathbb{P}(Z'_t = L)$. Then

$$\inf_{p_t \in \Delta'_{\mathcal{L}}} \left\{ \mathbb{E}_{\hat{v}_t \sim p_t} \mathbb{E}_{\delta_t} \mathbb{E}_{M_t} \inf_{f \in \mathcal{F}} \left\{ S_{f,-t} + 2\delta_t e^T_{f(x_t)} M_t \right\} \right\} \tag{4.49}$$

$$\geqslant \inf_{Z'_t \in \Delta_{\{0,L\}} : \mathbb{P}(Z'_t = L) \leqslant d2^d/L} \mathbb{E}_{\epsilon_t} \mathbb{E}_{Z'_t} \inf_{f \in \mathcal{F}} \left\{ S_{f,-t} + 2Z'_t e^T_{f(x_t)} \epsilon_t \right\} \tag{4.50}$$

$$= \inf_{a : 0 \leqslant a \leqslant d2^d/L} \left((1-a) \inf_{f \in \mathcal{F}} S_{f,-t} + a \mathbb{E}_{\epsilon_t} \inf_{f \in \mathcal{F}} \left\{ S_{f,-t} + 2L e^T_{f(x_t)} \epsilon_t \right\} \right) \tag{4.51}$$

Note that the infimum in (4.51) is achieved at $a = d2^d/L$. Indeed, it is enough to prove that

$$\inf_{f \in \mathcal{F}} S_{f,-t} \geqslant \mathbb{E}_{\epsilon_t} \inf_{f \in \mathcal{F}} \left\{ S_{f,-t} + 2L e^T_{f(x_t)} \epsilon_t \right\},$$

which is true because, if we denote by $f^* \coloneqq \operatorname{arginf}_{f\in\mathcal{F}} S_{f,-t}$, we have that

$$\mathbb{E}_{\epsilon_t} \inf_{f\in\mathcal{F}} \left\{S_{f,-t} + 2Le^T_{f(x_t)}\epsilon_t\right\} \leqslant \mathbb{E}_{\epsilon_t}\left[S_{f^*} + 2Le^T_{f^*(x_t)}\epsilon_t\right] = S_{f^*} + 2Le^T_{f^*(x_t)}\mathbb{E}_{\epsilon_t}[\epsilon_t] = S_{f^*}.$$

Thus, we can lower bound (4.49) as follows:

$$\inf_{p_t\in\Delta'_{\mathcal{L}}} \left\{\mathbb{E}_{\hat{v}_t\sim p_t}\mathbb{E}_{\delta_t}\mathbb{E}_{M_t} \inf_{f\in\mathcal{F}} \left\{S_{f,-t} + 2\delta_t e^T_{f(x_t)} M_t\right\}\right\} \geqslant \mathbb{E}_{\epsilon_t}\mathbb{E}_{Z_t} \inf_{f\in\mathcal{F}} \left\{S_{f,-t} + 2Z_t e^T_{f(x_t)}\epsilon_t\right\},$$

where $Z_t \in \{0, L\}$ and is a random variable which is equal to L with probability $d2^d/L$ and 0 otherwise. Therefore,

$$\min_{\tilde{v}_t} \mathbb{E}_{\hat{a}_t\sim q_t, X_t}\left\{\tilde{v}_t(\hat{a}_t) + \mathbf{Rel}^{\mathrm{BF}}(I_{1:t})\right\} \tag{4.52}$$

$$\geqslant \mathbb{E}_{\rho_t} \inf_{p_t\in\Delta'_{\mathcal{L}}} \left\{\mathbb{E}_{\hat{v}_t\sim p_t}\mathbb{E}_{\delta_t} \inf_{f\in\mathcal{F}} \left\{S_{f,-t} + 2\delta_t e^T_{f(x_t)}\hat{v}_t\right\}\right\} - \frac{(n-(t-1))d2^{d+1}}{L} \tag{4.53}$$

$$\geqslant \mathbb{E}_{\rho_t}\mathbb{E}_{\epsilon_t}\mathbb{E}_{Z_t} \inf_{f\in\mathcal{F}} \left\{S_{f,-t} + 2Z_t e^T_{f(x_t)}\epsilon_t\right\} - \frac{(n-(t-1))d2^{d+1}}{L}. \tag{4.54}$$

Now note that the lower bound in (4.54) holds for any x_t. Therefore, we can take the expectation on both sides with respect to x_t, so that

$$\mathbb{E}_{x_t} \min_{\tilde{v}_t} \mathbb{E}_{\hat{a}_t\sim q_t, X_t}\left\{\tilde{v}_t(\hat{a}_t) + \mathbf{Rel}^{\mathrm{BF}}(I_{1:t})\right\} \tag{4.55}$$

$$\geqslant \mathbb{E}_{(x,\epsilon,Z)_{(t:n)}} \inf_{f\in\mathcal{F}} \left\{S_{f,-t} + 2Z_t e^T_{f(x_t)}\epsilon_t\right\} - \frac{(n-(t-1))d2^{d+1}}{L} \tag{4.56}$$

$$= \mathbb{E}_{(x,\epsilon,Z)_{(t:n)}} \inf_{f\in\mathcal{F}} \left\{S_{f,-t} - 2Z_t e^T_{f(x_t)}\epsilon_t\right\} - \frac{(n-(t-1))d2^{d+1}}{L} \tag{4.57}$$

$$= \mathbf{Rel}^{\mathrm{BF}}(I_{1:(t-1)}). \tag{4.58}$$

This proves admissibility.

Regret bound. The final bound is finally given by

$$\mathbf{Rel}^{\mathrm{BF}}(\varnothing) = \mathbb{E}_{x_{1:n}}\mathbb{E}_{\epsilon_{1:n}}\mathbb{E}_{Z_{1:n}} \inf_{f\in\mathcal{F}} \left\{ -\sum_{s=1}^{n} 2e_{f(x_s)}^T \epsilon_s Z_s \right\} - \frac{nd2^{d+1}}{L} \tag{4.59}$$

$$= -2\mathbb{E}_{x_{1:n}}\mathbb{E}_{\epsilon_{1:n}}\mathbb{E}_{Z_{1:n}} \sup_{f\in\mathcal{F}} \left\{ \sum_{s=1}^{n} e_{f(x_s)}^T \epsilon_s Z_s \right\} - \frac{nd2^{d+1}}{L}, \tag{4.60}$$

and, thus, by Lemma 4.5, we have that

$$\mathbb{E}_{x_{1:n}}(\mathbf{Regret}) \leqslant -\mathbf{Rel}^{\mathrm{BF}}(\varnothing) = 2\mathbb{E}_{x_{1:n}}\mathbb{E}_{\epsilon_{1:n}}\mathbb{E}_{Z_{1:n}} \sup_{f\in\mathcal{F}} \left\{ \sum_{s=1}^{n} e_{f(x_s)}^T \epsilon_s Z_s \right\} + \frac{nd2^{d+1}}{L}. \tag{4.61}$$

This completes the proof. ■

Proof of Lemma 4.6

We prove this lemma by induction. For $n = 0$ the statement clearly holds true because both sides are zero. Now suppose the lemma holds for some $n \geqslant 0$. Then, for π Rademacher random variable, we have that

$$\mathbb{E}_{\epsilon_{1:n+1}}\left[\sup_M \sum_{s=1}^{n+1} \phi_s(M)^T \epsilon_s\right] = \mathbb{E}_{\epsilon_{1:n+1}}\left[\sup_M \sum_{s=1}^{n} \phi_s(M)^T \epsilon_s + \phi_{n+1}(M)^T \epsilon_{n+1}\right]$$

$$= \mathbb{E}_{\epsilon_{1:n+1}}\mathbb{E}_{\pi}\left[\sup_M \sum_{s=1}^{n} \phi_s(M)^T \epsilon_s + \phi_{n+1}(M)^T \epsilon_{n+1}\pi\right]$$

$$= \mathbb{E}_{\epsilon_{1:n+1}}\left\{\frac{1}{2}\left[\sup_M \sum_{s=1}^{n} \phi_s(M)^T \epsilon_s + \phi_{n+1}(M)^T \epsilon_{n+1}\right] + \right.$$

$$\left. + \frac{1}{2}\left[\sup_{M'} \sum_{s=1}^{n} \phi_s(M')^T \epsilon_s - \phi_{n+1}(M')^T \epsilon_{n+1}\right]\right\}$$

$$= \mathbb{E}_{\epsilon_{1:n+1}}\left[\sup_{M,M'} \sum_{s=1}^{n} \left(\frac{\phi_s(M) + \phi_s(M')}{2}\right)^T \epsilon_s + \left(\frac{\phi_{n+1}(M) - \phi_{n+1}(M')}{2}\right)^T \epsilon_{n+1}\right]$$

$$\overset{(a)}{=} \mathbb{E}_{\epsilon_{1:n}}\left[\sup_{M,M'} \sum_{s=1}^{n} \left(\frac{\phi_s(M) + \phi_s(M')}{2}\right)^T \epsilon_s + \frac{1}{2}\|\phi_{n+1}(M) - \phi_{n+1}(M')\|_1\right]$$

$$\overset{(b)}{\leqslant} \mathbb{E}_{\epsilon_{1:n}}\left[\sup_{M,M'} \sum_{s=1}^{n} \left(\frac{\phi_s(M) + \phi_s(M')}{2}\right)^T \epsilon_s + \frac{1}{2}\|\psi_{n+1}(M) - \psi_{n+1}(M')\|_1\right]$$

$$\overset{(c)}{=} \mathbb{E}_{\epsilon_{1:n}}\mathbb{E}_{\delta_{n+1}}\left[\sup_{M,M'} \sum_{s=1}^{n} \left(\frac{\phi_s(M) + \phi_s(M')}{2}\right)^T \epsilon_s + \left(\frac{\psi_{n+1}(M) - \psi_{n+1}(M')}{2}\right)^T \delta_{n+1}\right]$$

$$\overset{(d)}{=} \mathbb{E}_{\epsilon_{1:n}}\mathbb{E}_{\delta_{n+1}}\mathbb{E}_{\pi}\left[\sup_M \sum_{s=1}^{n} \phi_s(M)^T \epsilon_s + \psi_{n+1}(M)^T \delta_{n+1}\pi\right]$$

$$= \mathbb{E}_{\delta_{n+1}}\mathbb{E}_{\epsilon_{1:n}}\left[\sup_M \sum_{s=1}^{n} \phi_s(M)^T \epsilon_s + \psi_{n+1}(M)^T \delta_{n+1}\right]$$

$$\overset{(e)}{\leqslant} \mathbb{E}_{\delta_{n+1}}\mathbb{E}_{\delta_{1:n}}\left[\sup_M \sum_{s=1}^{n} \psi_s(M)^T \delta_s + \psi_{n+1}(M)^T \delta_{n+1}\right]$$

$$= \mathbb{E}_{\delta_{1:n+1}}\left[\sup_M \sum_{s=1}^{n+1} \psi_s(M)^T \delta_s\right],$$

where step (a) follows from the fact that it must be the case that whenever the j-th entry of ϵ_{n+1} is $+1$, the j-th entry of $\phi_{n+1}(M)$ must be larger that the j-th entry of $\phi_{n+1}(M')$ and whenever the j-th entry of ϵ_{n+1} is -1, the j-th entry of $\phi_{n+1}(M)$ must be smaller that the j-th entry of $\phi_{n+1}(M')$. This must hold because otherwise M and M' could not be the maximizers (because we can just swap the j-th entries and increase the objective). Step (b) follows from the assumption; step (c) holds for the same reason as step (a); step (d) follows from reversing the above steps (applied to ψ_{n+1}), and step (e) follows from the induction hypothesis. ■

Proof of Corollary 4.3

We apply Lemma 4.6 with $\mathcal{M} = \mathcal{F}$, $M = f$, $\phi_s(f) = \gamma(f(x_s))$ and $\psi_s(f) = Cf(x_s)$. Given the assumption on γ, we have that $\|\gamma(f(x_s)) - \gamma(f'(x_s))\|_1 \leqslant C\|f(x_s) - f'(x_s)\|_1$ for some $C > 0$, i.e., the conditions of Lemma 4.6 hold, yielding equation (4.25). ■

Proof of Proposition 4.3

We prove this proposition similarly to Proposition 4.1. As shown in Corollary 4.3, in order to bound $\mathbb{E}_{x_{1:n}}(\textbf{Regret})$, we need to bound the quantity

$$\mathbb{E}_{x_{1:n}}\mathbb{E}_{\delta_{1:n}}\mathbb{E}_{Z_{1:n}}\left[\sup_{f\in\mathcal{F}}\sum_{s=1}^{n} f(x_s)^T\delta_s Z_s\right].$$

Denote by $z_f \coloneqq \sum_{s=1}^{n} f(x_s)^T\delta_s Z_s$. Then, for $\lambda > 0$, we have that

$$
\begin{aligned}
e^{\lambda \mathbb{E}_{x_{1:n},\delta_{1:n},Z_{1:n}}[\sup z_f]} &\overset{(a)}{\leqslant} \mathbb{E}_{x_{1:n},\delta_{1:n},Z_{1:n}}\big[e^{\lambda \sup z_f}\big] \\
&\overset{(b)}{=} \mathbb{E}_{x_{1:n},\delta_{1:n},Z_{1:n}}\big[\sup e^{\lambda z_f}\big] \\
&\overset{(c)}{\leqslant} \sum_{f\in\mathcal{F}} \mathbb{E}_{x_{1:n},\delta_{1:n},Z_{1:n}}\big[e^{\lambda z_f}\big] \\
&= \sum_{f\in\mathcal{F}} \mathbb{E}_{x_{1:n},\delta_{1:n},Z_{1:n}}\left[\prod_{s=1}^{n} e^{\lambda f(x_s)^T \delta_s Z_s}\right] \\
&\overset{(d)}{=} \sum_{f\in\mathcal{F}} \prod_{s=1}^{n} \mathbb{E}_{x_s,\delta_s,Z_s}\big[e^{\lambda f(x_s)^T \delta_s Z_s}\big],
\end{aligned}
$$

where (a) follows from Jensen's inequality, (b) follows from the monotonicity of the exponential function, (c) follows from the fact that $|\mathcal{F}| < \infty$, and (d) follows from independence. Moreover, note that, since $\mathbb{E}[f(x_s)^T \delta_s Z_s] = 0$ and, for $k \geqslant 2$,

$$
\begin{aligned}
\mathbb{E}[(f(x_s)^T \delta_s Z_s)^k] &\leqslant \mathrm{Var}[f(x_s)^T \delta_s Z_s](dL)^{k-2} \\
&= \mathrm{Var}[f(x_s)^T \delta_s]\mathbb{E}[Z_s^2](dL)^{k-2} \\
&= \mathrm{Var}(f)(Ld2^d)(dL)^{k-2},
\end{aligned}
$$

we have that

$$
\mathbb{E}_{x_s,\delta_s,Z_s}\big[e^{\lambda f(x_s)^T \delta_s Z_s}\big] = 1 + \sum_{k=2}^{\infty} \frac{\lambda^k \mathbb{E}[(f(x_s)^T \delta_s Z_s)^k]}{k!} \leqslant 1 + \frac{\mathrm{Var}(f) Ld2^d}{(dL)^2}(e^{\lambda dL} - \lambda dL - 1).
$$

Therefore, letting $\lambda = \mu/\sqrt{n}$ for some $\mu > 0$ (to be chosen later), the overall bound is

$$\begin{aligned}
\mathbb{E}_{x_{1:n},\delta_{1:n},Z_{1:n}}[\sup z_f] &\leqslant \frac{\sqrt{n}}{\mu}\log\left[\sum_{f\in\mathcal{F}}\left(1+\frac{\mathrm{Var}(f)Ld2^d}{(dL)^2}(e^{\lambda dL}-\lambda dL-1)\right)^n\right] \\
&\overset{(a)}{\leqslant} \frac{\sqrt{n}}{\mu}\log\left[\sum_{f\in\mathcal{F}}\left(1+\frac{\mathrm{Var}(f)Ld2^d}{(dL)^2}(\lambda dL)^2\right)^n\right] \\
&= \frac{\sqrt{n}}{\mu}\log\left[\sum_{f\in\mathcal{F}}\left(1+\frac{\mathrm{Var}(f)Ld2^d\mu^2}{n}\right)^n\right] \\
&\overset{(b)}{\leqslant} \frac{\sqrt{n}}{\mu}\log\left[\sum_{f\in\mathcal{F}}e^{\mathrm{Var}(f)Ld2^d\mu^2}\right] \\
&\leqslant \frac{\sqrt{n}}{\mu}\log\left[|\mathcal{F}|e^{\mathrm{Var}(f^*)Ld2^d\mu^2}\right],
\end{aligned}$$

where (a) follows from the inequality $e^x - x - 1 \leqslant x^2$ for $x < 1.79$, and (b) follows from the inequality $(1 + a/n)^n \leqslant e^a$ for $a \geqslant 0$. Now, minimizing the right hand side with respect to μ, we obtain $\mu^* = \sqrt{\frac{\log|\mathcal{F}|}{\mathrm{Var}(f^*)Ld2^d}}$, leading to

$$\mathbb{E}_{x_{1:n},\epsilon_{1:n},Z_{1:n}}[\sup z_f] \leqslant 2\sqrt{n\log|\mathcal{F}|\mathrm{Var}(f^*)Ld2^d}.$$

Note that in order for this bound to hold, we must have (according to (a)) that

$$\lambda^* dL < 1.79 \iff \sqrt{\frac{\log|\mathcal{F}|}{n\mathrm{Var}(f^*)Ld2^d}}dL < 1.79 \iff L < \frac{(1.79^2)n2^d\mathrm{Var}(f^*)}{d\log(|\mathcal{F}|)}. \tag{4.62}$$

Therefore, by Lemma 4.5, the final bound is given by

$$\mathbb{E}_{x_{1:n}}(\mathbf{Regret}) \leqslant -\mathbf{Rel}^{\mathrm{BF}}(\varnothing) \leqslant 8\sqrt{n\log(|\mathcal{F}|)\mathrm{Var}(f^*)Ld2^d} + \frac{nd2^{d+1}}{L}.$$

Setting $L^* = \frac{1}{2^{2/3}}\left(\frac{nd2^d}{\log(|\mathcal{F}|)\mathrm{Var}(f^*)}\right)^{1/3}$, i.e., minimizing the upper bound and satisfying equation (4.62), the final bound becomes

$$\mathbb{E}_{x_{1:n}}(\mathbf{Regret}) \leqslant 2^{14/3}\left(nd2^d\right)^{2/3}\left(\log(|\mathcal{F}|)\mathrm{Var}(f^*)\right)^{1/3},$$

which holds whenever $n \geqslant 4(d2^d)^2 \log(|\mathcal{F}|)\mathrm{Var}(f^*)$ (since we require $L \geqslant d2^d$). This completes the proof. ■

Proof of Proposition 4.4

Note that, given that $\mathcal{F} = \{f : \mathcal{X} \to \mathcal{A} : f(x) = \tau(Mx)\}$, reasoning conditional on $x_{1:n}$, the quantity that we want to bound is

$$\mathbb{E}_{\delta_{1:n}} \mathbb{E}_{Z_{1:n}} \left[\sup_M \sum_{s=1}^n \tau(Mx_s)^T \delta_s Z_s \right].$$

In order to do this, we define the function

$$g(x) = Mx + \lambda \begin{pmatrix} \mathbb{1}\{[Mx]_1 \geqslant 0\} \\ \mathbb{1}\{[Mx]_2 \geqslant 0\} \\ \vdots \\ \mathbb{1}\{[Mx]_d \geqslant 0\} \end{pmatrix}$$

for some $\lambda > 0$ to be chosen later. The function g shifts the nonnegative entries of Mx so that they are at least λ away from zero. Clearly, given x_s, $\tau(Mx_s)^T \delta_s Z_s \stackrel{d}{=} \tau(g(x_s))^T \delta_s Z_s$ for any s, where the equality holds in distribution. Thus, it is enough to bound

$$\mathbb{E}_{Z_{1:n}} \mathbb{E}_{\delta_{1:n}} \left[\sup_M \sum_{s=1}^n \tau(g(x_s))^T \delta_s Z_s \right].$$

Now, notice that for each s and g, g', we have

$$\left| [\tau(g(x_s))]_j - [\tau(g'(x_s))]_j \right| \leqslant \frac{1}{\lambda} \left| [g(x_s)]_j - [g'(x_s)]_j \right| \tag{4.63}$$

for each coordinate $j = 1, \ldots, d$. The reason equation (4.63) holds is because if $[g(x_s)]_j$ and $[g'(x_s)]_j$ have the same sign, the left-hand side is zero while the right-hand side is nonnegative; if $[g(x_s)]_j$ and $[g'(x_s)]_j$ have opposite signs, the left-hand

side is one while the right-hand side is at least one because $\lambda \leqslant |[g(x_s)]_j - [g'(x_s))]_j|$, given the definition of g. Therefore, $\tau(g(\cdot))$ satisfies the conditions of Corollary 4.1 with $C = 1/\lambda$, so that

$$\begin{aligned}
&\mathbb{E}_{Z_{1:n}}\mathbb{E}_{\delta_{1:n}}\left[\sup_M \sum_{s=1}^n \tau(g(x_s))^T \delta_s Z_s\right] \\
&\leqslant \frac{1}{\lambda}\left(\mathbb{E}_{Z_{1:n}}\mathbb{E}_{\delta_{1:n}}\left[\sup_M \sum_{s=1}^n g(x_s)^T \delta_s Z_s\right]\right) \\
&\overset{(a)}{\leqslant} \frac{1}{\lambda}\left(\mathbb{E}_{Z_{1:n}}\mathbb{E}_{\delta_{1:n}}\left[\sup_M \sum_{s=1}^n (Mx_s)^T \delta_s Z_s\right] + \mathbb{E}_{Z_{1:n}}\mathbb{E}_{\delta_{1:n}}\left[\sup_{y_s \in \{0,\lambda\}^d} \sum_{s=1}^n y_s^T \delta_s Z_s\right]\right),
\end{aligned}$$

where (a) follows from splitting the supremum into two parts and taking the supremum in the second term over all possible vectors. We can now bound the second term on the right-hand side by Massart finite lemma (by reasoning conditional on Z_s), yielding the upper bound $d\lambda\sqrt{2n\mathbb{E}[Z_s^2]\log(2^d)}$; while for the first term we can follow similar steps as in the proof of Lemma 4.3, yielding $CN\sqrt{dn\mathbb{E}[Z_s^2]\max_{s,i}([x_s]_i)^2}$. Therefore, we have

$$\begin{aligned}
&\mathbb{E}_{Z_{1:n}}\mathbb{E}_{\delta_{1:n}}\left[\sup_M \sum_{s=1}^n \tau(g(x_s))^T \delta_s Z_s\right] \\
&\leqslant \frac{1}{\lambda}\left(CN\sqrt{dn\mathbb{E}[Z_s^2]\max_{s,i}([x_s]_i)^2} + \lambda d\sqrt{2n\mathbb{E}[Z_s^2]\log(2^d)}\right) \\
&= \frac{1}{\lambda}\sqrt{nd\mathbb{E}[Z_s^2]}\left(CN\sqrt{\max_{s,i}([x_s]_i)^2} + d\lambda\sqrt{2\log(2)}\right),
\end{aligned}$$

Setting $\lambda = 1$, and since $\mathbb{E}[Z_s^2] = Ld2^d$, we have

$$\mathbb{E}_{Z_{1:n}}\mathbb{E}_{\delta_{1:n}}\left[\sup_M \sum_{s=1}^n \tau(g(x_s))^T \delta_s Z_s\right] \leqslant \sqrt{nLd^2 2^d}\left(CN\sqrt{\max_{s,i}([x_s]_i)^2} + d\sqrt{2\log(2)}\right).$$

Given Corollary 4.4, the expected regret can be upper bounded by

$$4\sqrt{nLd^2 2^d}\left(CN\mathbb{E}_{x_{1:n}}\left[\sqrt{\max_{s,i}([x_s]_i)^2}\right] + d\sqrt{2\log(2)}\right) + \frac{nd2^{d+1}}{L}.$$

Minimizing this quantity with respect to L, we obtain

$$L^* = \left(\frac{n2^d}{\left(CN\mathbb{E}_{x_{1:n}}\left[\sqrt{\max_{s,i}([x_s]_i)^2}\right] + d\sqrt{2\log(2)}\right)^2}\right)^{1/3},$$

yielding the desired upper bound in (4.33), i.e.

$$\mathbb{E}_{x_{1:n}}(\textbf{Regret}) \leqslant 2^{8/3} d(n2^d)^{2/3}\left(CN\mathbb{E}_{x_{1:n}}\left[\sqrt{\max_{s,i}([x_s]_i)^2}\right] + d\sqrt{2\log(2)}\right)^{2/3},$$

which holds whenever $n \geqslant d^3(2^d)^2\left(CN\mathbb{E}_{x_{1:n}}\left[\sqrt{\max_{s,i}([x_s]_i)^2}\right] + d\sqrt{2\log(2)}\right)^2$ (since we require $L \geqslant d2^d$). This completes the proof. ■

Proof of Corollary 4.5

We prove this corollary similarly to Corollary 4.2. By Proposition 4.4, we have that

$$\mathbb{E}_{x_{1:n}}(\textbf{Regret}) \leqslant 2^{8/3} d(n2^d)^{2/3}\left(CN\mathbb{E}_{x_{1:n}}\left[\sqrt{\max_{s,i}([x_s]_i)^2}\right] + d\sqrt{2\log(2)}\right)^{2/3}.$$

whenever $n \geqslant d^3(2^d)^2\left(CN\mathbb{E}_{x_{1:n}}\left[\sqrt{\max_{s,i}([x_s]_i)^2}\right] + d\sqrt{2\log(2)}\right)^2$. Therefore, it is enough to bound $\mathbb{E}\left[\sqrt{\max_{s,i}([x_s]_i)^2}\right]$. Following similar steps as in Corollary 4.2, we obtain that

$$\mathbb{E}\left[\sqrt{\max_{s,i}([x_s]_i)^2}\right] \leqslant \sqrt{\frac{\log(nN)}{2}} + \eta,$$

leading to the final bound

$$\mathbb{E}_{x_{1:n}}(\textbf{Regret}) \leqslant 2^{8/3} d(n2^d)^{2/3}\left(CN\left(\sqrt{\frac{\log(nN)}{2}} + \eta\right) + d\sqrt{2\log(2)}\right)^{2/3}.$$

This completes the proof. ■

Proof of Lemma 4.7

For $i \in \{0, 1, \ldots, 2^d\}$, define:

$$\psi_i \coloneqq \sup_{f \in \mathcal{F}} \left\{ \sum_{s=1}^{t-1} e_{f(x_s)}^T \hat{v}_s + L e_{f(x_t)}^T e_i + \sum_{s=t+1}^{n} 2 e_{f(x_s)}^T \epsilon_s Z_s \right\},$$

with the convention that $e_0 = \mathbf{0}$. Note that, with this notation we can rewrite $q_t^*(\rho_t)$ as follows:

$$\begin{aligned} q_t^*(\rho_t) &= \operatorname*{argsup}_{q \in \Delta_{\mathcal{A}}} \inf_{p_t \in \Delta'_{\mathcal{L}}} \mathbb{E}_{\hat{v} \sim p_t} \left[q^T \hat{v} + R(x_{1:t}, \hat{v}_{1:t-1}, \hat{v}, \rho_t) \right] \\ &\overset{(a)}{=} \operatorname*{argsup}_{q \in \Delta_{\mathcal{A}}} \inf_{p_t \in \Delta'_{\mathcal{L}}} \sum_{i=1}^{2^d} \left[p_t(i) q(i) L - p_t(i) \psi_i \right] - p_t(0) \psi_0 \\ &= \operatorname*{argsup}_{q \in \Delta_{\mathcal{A}}} \inf_{p_t \in \Delta'_{\mathcal{L}}} \sum_{i=1}^{2^d} p_t(i) \left[q(i) L - \psi_i \right] - p_t(0) \psi_0, \end{aligned}$$

where (a) follows from the fact that $\hat{v} \in \{Le_i : i \in [2^d]\} \cup \{\mathbf{0}\}$ and the constant $(n-t)d2^{d+1}/L$ does not change the maximizer in $\Delta_{\mathcal{A}}$. Note that each ψ_i can be computed with a single oracle access, for a total of $2^d + 1$ accesses. Thus, we can assume that they are given. We now show how to compute the maximizer of equation (4.30).

For each given q, the infimum over p_t can be characterized as follows. With the notation $z_i = q(i)L - \psi_i$ and $z_0 = -\psi_0$, we can rewrite $q_t^*(\rho_t)$ as follows

$$q_t^*(\rho_t) = \operatorname*{argsup}_{q \in \Delta_{\mathcal{A}}} \inf_{p_t \in \Delta'_{\mathcal{L}}} \sum_{i=1}^{2^d} p_t(i) z_i + p_t(0) z_0.$$

Note that if we did not have any constraint on $p_t(i) \leqslant d/L$ for $i > 0$, then we would have put all the probability mass on the minimum of the z_i. Given the constraint, the best we can do is to put as much probability mass as allowed on the minimum

coordinate $\operatorname{argmin}_{i\in\{0,\dots,2^d\}} z_i$ and continue to the next smallest quantity. We repeat this until we reach the quantity z_0. At this point, the probability mass that we can put on coordinate 0 is unconstrained. Therefore, we can put all the remaining probability mass on this coordinate.

Let $z_{(1)}, z_{(2)}, \dots, z_{(2^d)}$, denote the ascending order of the z_i quantities for $i > 0$ (from smallest to largest). Moreover, let $k \in [2^d]$ be the largest index such that $z_{(k)} \leqslant z_0$. By the above reasoning, we have that, for given q, the infimum over $p_t \in \Delta'_{\mathcal{L}} = \{p \in \Delta_{\mathcal{L}} : p(i) \leqslant d/L \ \forall i \in [2^d]\}$ is equal to

$$\begin{aligned}
\inf_{p_t\in\Delta'_{\mathcal{L}}} \sum_{i=1}^{2^d} p_t(i)z_i + p_t(0)z_0 &= \sum_{s=1}^{k} \frac{d}{L} z_{(s)} + \left(1 - \frac{kd}{L}\right) z_0 \\
&= \sum_{s=1}^{k} \frac{d}{L} z_{(s)} + \left(1 - \sum_{s=1}^{k} \frac{d}{L}\right) z_0 \\
&= \sum_{s=1}^{k} \frac{d}{L} \left(z_{(s)} - z_0\right) + z_0 \\
&\overset{(a)}{=} \sum_{i=1}^{2^d} \frac{d}{L} \min\{z_i - z_0, 0\} + z_0 \\
&\overset{(b)}{=} -\sum_{i=1}^{2^d} \frac{d}{L} \max\{z_0 - z_i, 0\} + z_0,
\end{aligned}$$

where (a) follows from the fact that, if $s > k$, then $z_{(s)} > z_0$ and we are assigning probability mass zero to such $z_{(s)}$; while (b) follows from the fact that $\min\{x, 0\} = -\max\{-x, 0\}$. Therefore, using the notation $(x)^+ = \max\{x, 0\}$, we can further

rewrite $q_t^*(\rho_t)$ as follows

$$\begin{aligned} q_t^*(\rho_t) &= \operatorname*{argsup}_{q \in \Delta_{\mathcal{A}}} - \sum_{i=1}^{2^d} \frac{d}{L}(z_0 - z_i)^+ + z_0 \\ &\overset{(a)}{=} \operatorname*{arginf}_{q \in \Delta_{\mathcal{A}}} \sum_{i=1}^{2^d} \frac{d}{L}(z_0 - z_i)^+ + z_0 \\ &\overset{(b)}{=} \operatorname*{arginf}_{q \in \Delta_{\mathcal{A}}} \sum_{i=1}^{2^d} \frac{d}{L}(z_0 - z_i)^+ \\ &\overset{(c)}{=} \operatorname*{arginf}_{q \in \Delta_{\mathcal{A}}} \sum_{i=1}^{2^d} d \left(\frac{\psi_i - \psi_0}{L} - q(i) \right)^+, \end{aligned}$$

where (a) follows from the fact that the maximizer of the objective will be the same as the minimizer of the negative objective, (b) holds since z_0 is a constant and (c) follows from the fact that $1/L$ is monotone for $L \geqslant 0$. Now, let $\phi_i \coloneqq \frac{\psi_i - \psi_0}{L}$, so that

$$q_t^*(\rho_t) = \operatorname*{arginf}_{q \in \Delta_{\mathcal{A}}} \sum_{i=1}^{2^d} d \left(\phi_i - q(i) \right)^+.$$

Note that the latter can be minimized as follows. Consider any $i \in [2^d]$ such that $\phi_i \leqslant 0$. Then, for such i, it is optimal to set $q(i) = 0$ because $(\phi_i - q(i))^+ = 0$ no matter what. Consider now the indices i for which $\phi_i > 0$. For such indices, the objective will not increase if we set any $q(i) \geqslant \phi_i$. Thus, a minimizer can be obtained by successively assigning the probability as the minimum between the current remaining probability and some ϕ_i for which $\phi_i > 0$. We can keep doing this until we assign all of the probability mass. At the end, if there is any remaining probability mass, we can distribute it arbitrarily to any $q(i)$ such that $\phi_i > 0$ (because the objective will not increase). ■

5

Conclusion

The research presented in this thesis addresses the challenge of developing practical and near-optimal algorithms for sequential decision-making problems arising in online marketplaces, with a particular focus on e-commerce and rental platforms. This work revisits and expands upon existing models, such as the multi-item order fulfillment model, introducing novel dynamic policies for resource allocation. By leveraging techniques such as randomized fulfillment strategies, prophet inequalities, and subgradient methods, these policies have not only proven to achieve asymptotic optimality and strong approximation guarantees but also to provide robust solutions in a wide range of problem instances.

Collectively, the methodologies and algorithms developed in this dissertation offer new approaches for optimizing decision-making processes such as resource allocation and recommendations. For each of these decision problems, we have shown how optimization and applied probability theory can be used to design policies that are theoretically guaranteed to perform well. It is my aspiration that the techniques and insights from this research will inspire further exploration and find broader application in various online platforms and beyond, aiding managers and decision-makers in

navigating the complexities of today's digital marketplaces.

Bibliography

Acimovic, J. and Farias, V. F. (2019), "The Fulfillment-Optimization Problem," *Operations Research*, pp. 218–237.

Acimovic, J. and Graves, S. C. (2015), "Making better fulfillment decisions on the fly in an online retail environment," *Manufacturing & Service Operations Management*, 17, 34–51.

Acimovic, J. and Graves, S. C. (2017), "Mitigating Spillover in Online Retailing via Replenishment," *Manufacturing & Service Operations Management*, 19, 419–436.

Agarwal, A., Hsu, D., Kale, S., Langford, J., Li, L., and Schapire, R. (2014), "Taming the monster: A fast and simple algorithm for contextual bandits," in *International Conference on Machine Learning*, pp. 1638–1646, PMLR.

Alaei, S. (2014), "Bayesian Combinatorial Auctions: Expanding Single Buyer Mechanisms to Many Buyers," *SIAM Journal on Computing*, 43, 930–972.

Amil, A., Makhdoumi, A., and Wei, Y. (2022), "Multi-item order fulfillment revisited: Lp formulation and prophet inequality," *Working Paper, Duke University*.

Andrews, J. M., Farias, V. F., Khojandi, A. I., and Yan, C. M. (2019), "Primal–Dual Algorithms for Order Fulfillment at Urban Outfitters, Inc." *INFORMS Journal on Applied Analytics*, 49, 355–370.

Aouad, A. and Saban, D. (2020), "Online assortment optimization for two-sided matching platforms," *Available at SSRN 3712553*.

Aouad, A. and Saritaç, Ö. (2020), "Dynamic stochastic matching under limited time," in *Proceedings of the 21st ACM Conference on Economics and Computation*, pp. 789–790.

Arlotto, A. and Gurvich, I. (2019), "Uniformly bounded regret in the multisecretary problem," *Stochastic Systems*, 9, 231–260.

Arlotto, A., Keskin, I. N., and Wei, Y. (2023), "Online Demand Fulfillment Problem with Initial Inventory Placement: A Regret Analysis," *Available at SSRN 4666493*.

Asadpour, A., Wang, X., and Zhang, J. (2020), "Online resource allocation with limited flexibility," *Management Science*, 66, 642–666.

Ashlagi, I., Burq, M., Dutta, C., Jaillet, P., Saberi, A., and Sholley, C. (2019a), "Edge weighted online windowed matching," in *Proceedings of the 2019 ACM Conference on Economics and Computation*, pp. 729–742.

Ashlagi, I., Burq, M., Jaillet, P., and Manshadi, V. (2019b), "On matching and thickness in heterogeneous dynamic markets," *Operations Research*, 67, 927–949.

Auer, P., Cesa-Bianchi, N., Freund, Y., and Schapire, R. E. (1995), "Gambling in a rigged casino: The adversarial multi-armed bandit problem." *Foundations of Computer Science (FOCS)*.

Aveklouris, A., DeValve, L., Stock, M., and Ward, A. R. (2021), "Matching impatient and heterogeneous demand and supply," *arXiv preprint arXiv:2102.02710*.

Azar, P. D., Kleinberg, R., and Weinberg, S. M. (2014), "Prophet inequalities with limited information," in *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pp. 1358–1377.

Babaioff, M., Immorlica, N., and Kleinberg, R. (2007), "Matroids, secretary problems, and online mechanisms," in *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pp. 434–443.

Baek, J. and Ma, W. (2022), "Technical Note—Bifurcating Constraints to Improve Approximation Ratios for Network Revenue Management with Reusable Resources," *Operations Research*, 70, 2226–2236.

Balseiro, S., Golrezaei, N., Mahdian, M., Mirrokni, V., and Schneider, J. (2019), "Contextual Bandits With Cross-Learning," *NIPS*.

Balseiro, S. R. and Brown, D. B. (2019), "Approximations to stochastic dynamic programs via information relaxation duality," *Operations Research*, 67, 577–597.

Balseiro, S. R., Besbes, O., and Pizarro, D. (2023), "Survey of dynamic resource-constrained reward collection problems: Unified model and analysis," *Operations Research*.

Bastani, H. and Bayati, M. (2019), "Online Decision Making with High-Dimensional Covariates," *Operations Research*, 68, 276–294.

Bastani, H., Bayati, M., and Khosravi, K. (2020), "Mostly Exploration-Free Algorithms for Contextual Bandits," *Management Science*, 67, 1329–1349.

Bubeck, S., Cesa-Bianchi, N., and Kakade, S. (2012), "Towards Minimax Policies for Online Linear Optimization with Bandit Feedback," in *25th Annual Conference on Learning Theory*.

Cassel, A., Mannor, S., and Zeevi, A. (2018), "A General Approach to Multi-Armed Bandits Under Risk Criteria," in *Proceedings of the 17 ACM Conference on Computational Learning Theory (COLT)*.

Castro, F., Nazerzadeh, H., and Yan, C. (2020), "Matching queues with reneging: a product form solution," *Queueing Systems*, pp. 1–27.

Cesa-Bianchi, N. and Lugosi, G. (2006), *Prediction, learning, and games*, Cambridge university press.

Cesa-Bianchi, N., Freund, Y., Haussel, D., Helmbold, D. P., Schapire, R. E., and Warmuth, M. K. (1997), "How to use expert advice," *Journal of the ACM (JACM)*.

Chen, X., Feldman, J., Jung, S. H., and Kouvelis, P. (2022), "Approximation schemes for the joint inventory selection and online resource allocation problem," *Production and Operations Management*, 31, 3143–3159.

Cohen, M. C., Lobel, I., and Paes Leme, R. (2020), "Feature-Based Dynamic Pricing," *Management Science*, 66, 4921–4943.

Correa, J., Foncea, P., Hoeksma, R., Oosterwijk, T., and Vredeveld, T. (2019), "Recent developments in prophet inequalities," *ACM SIGecom Exchanges*, 17, 61–70.

Dani, V., Hayes, T., and Kakade, S. (2007), "The Price of Bandit Information for Online Optimization," *NIPS*.

Désir, A., Goyal, V., and Zhang, J. (2022), "Capacitated assortment optimization: Hardness and approximation," *Operations Research*, 70, 893–904.

DeValve, L., Wei, Y., Wu, D., and Yuan, R. (2023), "Understanding the value of fulfillment flexibility in an online retailing environment," *Manufacturing & service operations management*, 25, 391–408.

Dudik, M., Hsu, D., Kale, S., Karampatziakis, N., Langford, J., Reyzin, L., and Zhang, T. (2011), "Efficient optimal learning for contextual bandits," in *Proceedings of the 27th Conference on Uncertainty in Artificial Intelligence, UAI 2011*, p. 169.

Dutting, P., Feldman, M., Kesselheim, T., and Lucier, B. (2020), "Prophet inequalities made easy: Stochastic optimization by pricing nonstochastic inputs," *SIAM Journal on Computing*, 49, 540–582.

Elmachtoub, A. N. and Levi, R. (2016), "Supply chain management with online customer selection," *Operations Research*, 64, 458–473.

Feldman, J., Mehta, A., Mirrokni, V., and Muthukrishnan, S. (2009), "Online stochastic matching: Beating $1 - 1/e$," in *2009 50th Annual IEEE Symposium on Foundations of Computer Science*, pp. 117–126.

Feng, Y. and Niazadeh, R. (2020), "Batching and Optimal Multi-stage Bipartite Allocations," *Chicago Booth Research Paper*.

Feng, Y., Niazadeh, R., and Saberi, A. (2022), "Near-optimal bayesian online assortment of reusable resources," in *Proceedings of the 23rd ACM Conference on Economics and Computation*, pp. 964–965.

Foster, D. J., Rakhlin, A., David, S.-L., and Xu, Y. (2020), "Instance-Dependent Complexity of Contextual Bandits and Reinforcement Learning: A Disagreement-Based Perspective," *arXiv preprint arXiv:2010.03104*.

Freund, Y. and Schapire, R. E. (1997), "A decision-theoretic generalization of online learning and an application to boosting," *Journal of Computer and System Sciences*.

Golrezaei, N., Nazerzadeh, H., and Rusmevichientong, P. (2014), "Real-time optimization of personalized assortments," *Management Science*, 60, 1532–1551.

Harsha, P., Subramanian, S., and Uichanco, J. (2019), "Dynamic Pricing of Omnichannel Inventories," *Manufacturing & Service Operations Management*, 21, 47–65.

Jasin, S. and Sinha, A. (2015), "An LP-Based Correlated Rounding Scheme for Multi-Item Ecommerce Order Fulfillment," *Operations Research*, 63, 1336–1351.

Jiang, J., Ma, W., and Zhang, J. (2021), "Tight Guarantees for Multi-unit Prophet Inequalities and Online Stochastic Knapsack," *arXiv preprint arXiv:2107.02058*.

Johari, R., Kamble, V., and Kanoria, Y. (2021), "Matching while learning," *Operations Research*.

Keskin, B. and Birge, J. R. (2019), "Dynamic Selling Mechanisms for Product Differentiation and Learning," *Operations Research*, 67, 1136–1167.

Keskin, B. and Zeevi, A. (2017), "Chasing Demand: Learning and Earning in a Changing Environment," *Mathematics of Operations Research*, 42, 277–307.

Kleinberg, R. and Weinberg, S. M. (2012), "Matroid prophet inequalities," in *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, pp. 123–136.

Krengel, U. and Sucheston, L. (1978), "On semiamarts, amarts, and processes with finite value," *Probability on Banach spaces*, 4, 197–266.

Langford, J. and Zhang, T. (2007), "Epoch-Greedy algorithm for multi-armed bandits with side information," *Advances in Neural Information Processing Systems (NIPS 2007)*, 20, 1.

Lei, Y. M., Jasin, S., and Sinha, A. (2018), "Joint Dynamic Pricing and Order Fulfillment for E- commerce Retailers," *Manufacturing & Service Operations Management*, 20, 269–284.

Levi, R. and Radovanović, A. (2010), "Provably near-optimal LP-based policies for revenue management in systems with reusable resources," *Operations Research*, 58, 503–507.

Lo, I., Manshadi, V., Rodilitz, S., and Shameli, A. (2020), "Optimal Growth in Repeated Matching Platforms: Options versus Adoption," *arXiv preprint arXiv:2005.10731*.

Lobel, I., Paes Leme, R., and Vladu, A. (2018), "Multidimensional Binary Search for Contextual Decision-Making," *Operations Research*, 66, 1346–1361.

Ma, W. (2023), "Order-Optimal Correlated Rounding for Fulfilling Multi-Item E-Commerce Orders," *Manufacturing & Service Operations Management*.

Ma, W. and Simchi-Levi, D. (2020), "Algorithms for online matching, assortment, and pricing with tight weight-dependent competitive ratios," *Operations Research*, 68, 1787–1803.

Ma, W., Simchi-Levi, D., and Teo, C.-P. (2021), "On policies for single-leg revenue management with limited demand information," *Operations Research*, 69, 207–226.

Ma, Y., Rusmevichientong, P., Sumida, M., and Topaloglu, H. (2020), "An Approximation Algorithm for Network Revenue Management Under Nonstationary Arrivals." *Operations Research*, 68, 834–855.

Manshadi, V. and Rodilitz, S. (2020), "Online Policies for Efficient Volunteer Crowdsourcing," in *Proceedings of the 21st ACM Conference on Economics and Computation*, pp. 315–316.

Manshadi, V., Rodilitz, S., Saban, D., and Suresh, A. (2022), "Online Algorithms for Matching Platforms with Multi-Channel Traffic," *arXiv preprint arXiv:2203.15037*.

Manshadi, V. H., Gharan, S. O., and Saberi, A. (2012), "Online stochastic matching: Online actions based on offline statistics," *Mathematics of Operations Research*, 37, 559–573.

Mehta, A. et al. (2013), "Online matching and ad allocation," *Foundations and Trends® in Theoretical Computer Science*, 8, 265–368.

Niazadeh, R., Golrezaei, N., Wang, J., Susan, F., and Badanidiyuru, A. (2021), "Online Learning via Offline Greedy Algorithms: Applications in Market Design and Optimization," in *ACM Conference on Economics & Computation (EC)*.

Perchet, V., Rigollet, P., et al. (2013), "The multi-armed bandit problem with covariates," *The Annals of Statistics*, 41, 693–721.

Rakhlin, A. and Sridharan, K. (2015), "Hierarchies of Relaxations for Online Prediction Problems with Evolving Constraints," *arXiv preprint arXiv:1503.01212*.

Rakhlin, A. and Sridharan, K. (2016), "BISTRO: An Efficient Relaxation-Based Method for Contextual Bandits," *arXiv preprint arXiv:1602.02196*.

Rakhlin, S., Shamir, O., and Sridharan, K. (2012), "Relax and randomize: From value to algorithms," in *Advances in Neural Information Processing Systems*, pp. 2141–2149.

Samuel-Cahn, E. (1984), "Comparison of threshold stop rules and maximum for independent nonnegative random variables," *the Annals of Probability*, pp. 1213–1216.

Slivkins, A. (2019), *Introduction to Multi-Armed Bandits*, Foundations and Trends in Machine Learning.

Statista Research Department (2023), "Retail e-commerce sales worldwide from 2014 to 2027," `https://www.statista.com/statistics/379046/worldwide-retail-e-commerce-sales/`.

Statista Research Department (2024), "Revenue of the e-commerce industry in the U.S. 2018-2028," `https://www.statista.com/statistics/272391/us-retail-e-commerce-sales-forecast/#statisticContainer`.

Syrgkanis, V., Luo, H., Krishnamurthy, A., and Schapire, R. E. (2016), "Improved regret bounds for oracle based adversarial contextual bandits," in *29th Advances in Neural Information Processing Systems (NIPS)*.

Truong, V.-A. and Wang, X. (2019), "Prophet inequality with correlated arrival probabilities, with application to two sided matchings," *arXiv preprint arXiv:1901.02552*.

Wei, Y., Xu, J., and Yu, S. H. (2023), "Constant regret primal-dual policy for multi-way dynamic matching," in *Abstract Proceedings of the 2023 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, pp. 79–80.

Williamson, D. P. and Shmoys, D. B. (2011), *The design of approximation algorithms*, Cambridge university press.

Xu, P. J., Allgor, R., and Graves, S. C. (2009), "Benefits of Reevaluating Real-Time Order Fulfillment Decisions," *Manufacturing & Service Operations Management*, 11, 340–355.

Xu, Z., Zhang, H., Zhang, J., and Zhang, R. Q. (2020), "Online demand fulfillment under limited flexibility," *Management Science*, 66, 4667–4685.

Zhao, Y., Wang, X., and Xin, L. (2020), "Multi-Item Online Order Fulfillment in a Two-Layer Network," *Available at SSRN*.

Biography

Ayoub Amil is a fifth-year Ph.D. Candidate in the Decision Sciences Area (within the Department of Business Administration) at the Fuqua School of Business at Duke University, under the supervision of Professor Ali Makhdoumi and Professor Yehua Wei. His research interests lie in the broad area of sequential decision-making under uncertainty. Ayoub's research has been recognized at prestigious conferences such as the ACM EC '23 Conference, and he was a finalist for the RMP Jeff McGill Student Paper Award '22. After his Ph.D., Ayoub will join LinkedIn as a Senior Applied Research Data Scientist in 2024.

Ayoub earned his Bachelor of Science in Economics and Finance, graduating Cum Laude from Università degli Studi di Torino in 2016. After discovering his passion for Mathematics, Ayoub continued his studies with a Master of Science in Stochastics and Data Science, graduating Summa Cum Laude from Università degli Studi di Torino in 2019. Concurrently, Ayoub was part of the Allievi Honors Program (2014-2019) at Collegio Carlo Alberto, where he received a Diploma Allievi in Economics and a Master of Science in Statistics and Applied Mathematics with Distinction.